CRANFIELD UNIVERSITY

JIN LI

Simulation and Optimization of Integrated Maintenance Strategies for an Aircraft Assembly Process

School of Engineering

MSc by Research
Academic Year: 2012 - 2013

Supervisor: Tarapong Sreenuch
Antonios Tsourdos

November 2013

CRANFIELD UNIVERSITY

School of Engineering

MSc by Research

Academic Year 2012 - 2013

JIN LI

Simulation and Optimization of Integrated Maintenance Strategies for an Aircraft Assembly Process

Supervisor: Tarapong Sreenuch
Antonios Tsourdos

November 2013

ABSTRACT

In this thesis, the COMAC ARJ21 fuselage's final assembly process is used as a case study. High production rate (i.e. number of aircraft assembled per year) with reasonable cost is the overall aim in this example. The output of final assembly will essentially affect the prior and subsequent processes of the overall ARJ21 production. From the collected field data, it was identified that a number of disruptions (or bottlenecks) in the assembly sequence were caused by breakdowns and maintenance of the (semi-)automatic assembly machines like portable computer numerical control (CNC) drilling machine, rivet gun and overhead crane. The focus of this thesis is therefore on the maintenance strategies (i.e. Condition-Based Maintenance (CBM)) for these equipment and how they impact the throughput of the fuselage assembly process.

The fuselage assembly process is modelled and analysed by using agent-based simulation in this thesis. The agent approach allows complex process interactions of assembly, equipment and maintenance to be captured and empirically studied. In this thesis, the built network is modelled as the sequence of activities in each stage. Each stage is broken down into critical activities which are parameterized by activity lead-time and equipment used. CBM based models of uncertain degradation and imperfect maintenance are used in the simulation study. A scatter search is used to find multi-objective optimal solutions for the CBM regime, where the maintenance-related cost and production rate are the optimization objectives. In this thesis, in order to ease computation intensity caused by running multiple simulations during the optimization and to simplify a multi-objective formulation, multiple Min-Max weightings are applied to trace Pareto front. The empirical analysis reviews the trade-offs between the production rate and maintenance cost and how these objectives are influenced by the design parameters.

Keywords:

Aircraft Assembly, Condition Based Maintenance, Agent Based Simulation, Multi-Objective Optimization.

ACKNOWLEDGEMENTS

First of all, I would like to thank my employer, Commercial Aircraft Cooperation of China (COMAC), for sponsoring me the tuition fee of Cranfield University. Especially thanks to my department, Manufacturing Engineering Department from Shanghai Aircraft Manufacturing Company (the manufacturing centre of COMAC), and the HR department for recommending and giving me the opportunity to study here in Cranfield University.

Second, I would like to thank China Scholarship Council (CSC), who supplies my living expenses during my study period in UK. With the help of CSC, I can focus on my study here and do not need to worry too much about my daily expenses.

My utmost thanks to my academic supervisors Dr. Tarapong Sreenuch and Prof. Antonios Tsourdos for their valuable guidance and encouragement during the research. In particular thanks to Dr. Tarapong Sreenuch. I really appreciate his advice and supervision and it is an honour to work under his supervision.

In special, I extend my deepest thanks to my parents and friends who always care about my life oversea. And also I am thankful to my colleagues in China for they still remember me during my study here. Last but not least, thanks to my colleagues and also classmates in Cranfield for their companionship in the past one year.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CBM	Condition Based Maintenance
ABS	Agent Based Simulation
DES	Discrete Event Simulation
MTBF	Mean Time Before Failure
MRT	Mean Response Time
MMT	Mean Maintenance Time
RUL	Remaining Useful Life
CM	Condition Monitoring

1 INTRODUCTION AND LITERTATURE REVIEW

Nowadays, aircraft manufacturers are operating in a global competitive environment. Increasing production rate and reducing costs are the key drivers in aircraft manufacturing. In order to meet the required production rate, (semi-) automatic assembly machines (e.g. Flexible Drilling Head [1], GRAWDE (Gear Rib Automated Wing Drilling Equipment), HAWED (Horizontal Automated Wing Drilling Equipment) [2]) have increasingly being used in the aircraft assembly line. These machines can deliver significant productivity gains on the shop floor by reducing the manual multi-step processes and overcoming the restricted worker access [3]. The production throughput will very much depend on the operational availability of these (semi-)automatic machines [4]. Thus, machine breakdowns and maintenance are a major cause of bottlenecks in the assembly line. How to manage these machines in an efficient and cost-effective way to maximize the overall product rate is a challenge to the aircraft manufacturers [2].

Maintenance involves fixing when equipment becomes out of order (corrective maintenance) and also includes performing routine actions which will keep the equipment in working in order or prevent failures from arising (i.e. preventive maintenance) [5, 6]. A maintenance strategy in general includes identification of parameters, inspection methods, plan execution and repair [7, 8]. In the recent decade, Condition Based Maintenance (CBM) has increasingly being integrated as part of the manufacturing system [4, 9, 10, 11, 12]. Its goal is to minimize unscheduled downtime and shift towards a more forward-looking approach by monitoring deterioration of equipment conditions. Examples of integrated CBM in manufacturing system are found in the areas of measurement equipment [13], plastic injection [14], plastic yoghurt pots [15], food and drink industry [16], PBL (Performance-Based Logistics) contracts [17] and generic stochastically deteriorating systems [18]. In these examples, it has been shown that CBM can potentially improve the overall cost and production rate of the manufacturing systems by increasing the machine availability while reducing the maintenance cost [17, 18].

In many cases, manufacturing for an example where many high-value assets (machines) are part of it, is impractical and economically not feasible to experiment different manufacturing processes based on the real objects [9]. The simulation approach allows complex process interactions of assembly, equipment and maintenance to be captured and empirically studied in a virtual environment without having to build a real manufacturing system. Agent-Based Simulation (ABS) and Discrete Event Simulation (DES) are used in the manufacturing domain. ABS is based on the dynamic interaction of entities involved in the process. Examples of the ABS are autonomic manufacturing execution system [19] and intelligent manufacturing (e.g. enterprise integration and collaboration, manufacturing process planning and scheduling) [20]. DES is on the other hand based on a fixed sequence of operations or process being performed over entities [9] . It is more widely adopted in manufacturing as the manufacturing or assembly processes (i.e. fixed sequence of operations) can be naturally captured [9]. The important aspects like quality, cost and time can be simulated and analysed which provide the basis in Manufacturing System Development (MSD) and Product Realization Process (PRP) [21].

In simulation, a model comprises several input variables or model parameters such as scheduling properties, process lead time and machine reliability. MSD or PRP aim to find optimal controllable parameters that will result in the most desirable outputs of the process, e.g. maximum production rate and minimum maintenance cost. To find an optimal solution, the simulation is iterated until the most optimal combination of variables is found; at each iteration the controllable variables are adjusted, the model is simulated and the simulation output is then evaluated against the design objectives [22, 23]. Evolutionary techniques (e.g. scatter search metaheuristic, genetic algorithms) are often applied to solve difficult simulation optimization problems [24, 25, 26].

In this thesis, CBM is exploited as part of a design solution for a (semi-)automatic aircraft assembly process that demands high production rate and until now there is no studies of CBM in an aircraft assembly process reported in the literature. This and the simulation optimization of a CBM integrated aircraft assembly process model will be the contribution of this thesis. In this study, Commercial Aircraft Cooperation of China (COMAC) ARJ21 regional jet final assembly is used as a what-if representative example to illustrate the impact of CBM on the aircraft assembly process.

2 AIM AND OBJECTIVES

2.1 Aim

The research aims to investigate how Condition-Based Maintenance (CBM) can be applied into aircraft assembly processes by using simulation and optimization to improve the production rate and balance the maintenance cost as well.

2.2 Objectives

1. To develop an Agent-Based Simulation (ABS) model of an aircraft assembly process.
2. To develop equipment's degradation and maintenance (i.e. Condition-Based) model.
3. By simulation and multi-objective optimization, to perform trade-off analysis of production loss and maintenance cost.

2.3 Contributions

The main contribution of this thesis is the application of Condition Based Maintenance (CBM) in an aircraft assembly process. Based on Pareto optimal solutions, the trade-off analysis provide an insight how CBM affects the performance of the aircraft assembly system in terms of production rate and maintenance cost. In this thesis, a multiple sampling weighted min-max approach is used to address the limitations in the simulation tool and computation resources.

The CBM enabled aircraft assembly system is modelled by Agent Based Simulation (ABS). In this case, self-aware (or active) properties of an entity (e.g. condition monitoring and triggering of maintenance orders) and complex active interactions between entities (e.g. machine, maintenance, process) are straightforwardly captured using the ABS framework. This contrasts to the traditional Discrete Event Simulation (DES) where the self-aware active properties have to be determined by the system, and hence unintuitively being passive. Therefore, ABS of CBM enable aircraft assembly system is therefore another contribution of this thesis.

2.4 Publications

Parts of this thesis have been published by the author:

- Li, J., T. Sreenuch and A. Tsourdos (2013). Condition Based Maintenance Optimization of an Aircraft Assembly Process Considering Multiple Objectives. In press: *ISRN Aerospace Engineering*.
- Sreenuch, T., J. Li and A. Tsourdos (2013). Simulation Optimization of Condition Based Maintenance for an Aircraft Assembly Process. Submitted to: *International Journal of Manufacturing Engineering*.

2.5 Thesis Outline

The thesis is organized as follows. Section 1 describes the background and motivation of the research by means of literature review. Section 2 gives the aim, objective and contributions of the research. Section 3 introduced an aircraft assembly process which is also the case study of this thesis and then identifies the performance bottlenecks in the process. Section 4 introduces the degradation process of machines at the beginning and then introduces two main types of maintenance strategies. CBM is the strategy be applied in this thesis and the advantage of it is described at the end of this section. Section 5 discusses the approaches of modelling and introduces the structure of this Agent Based Simulation model in detail. Section 6 describes a multi-objective simulation optimization approach and defines the objective functions in this thesis. Section 7 analyses the results from optimization and discusses the trade-off between two competing objectives. The meaning of Monte Carlo simulation is also described by means of illustrating the simulation data. Section 8 gives the conclusion of this thesis and suggests the future work.

3 AIRCRAFT ASSEMBLY PROCESSES

3.1 An Overview of COMAC and ARJ21

The Commercial Aircraft Corporation of China, Ltd. (COMAC) is a state-owned company which functions as the main vehicle in implementing large passenger aircraft programs in China. The main products include trunk liner program C919 and regional jet program ARJ21.

ARJ21, short for Advanced Regional Jet for the 21st Century, is a new type of turbofan short/medium large regional jet that is designed and manufactured in China with own independent intellectual property rights. The range of the standard ARJ21 is 2,225 km, which is mainly for meeting the operation requirements of hub-spoke routes. The maximum take-off weight of the aircraft is 40,500 kg, the maximum operating altitude 11,900 m, and the maximum range 3,700 km. Two CF34-10A engines are mounted on the rear of the aircraft. There are 78 seats in a dual-class configuration and 90 seats in a full economy class configuration. Its economic life is designed to be 60000 flying hours/20 calendar years [27].

The ARJ21 program is now in the ongoing certification process and is currently in transition from development stage to batch serial production. The ARJ21 has been received more than three hundreds orders as of 2013. COMAC has planned to increase its production rate to 30 aircrafts per year by 2015. However, at this state, the production of ARJ21 is heavily reliance on manual processes and inevitably limited to 1-2 aircrafts per year. To meet the delivery target (i.e. 30 aircrafts per year) while maintaining quality and cost effectiveness, the manufacturing and assembly processes of the ARJ21 have to be less of manual work, but more automated by adopting the concept of (semi-)automatic assembly process.

Similar to other integrated aircraft manufacturing networks like B777, B787, A340 and A380, the main structure components of the ARJ21 are manufactured and assembled across China by three other ARJ21 consortium members (Tier-1) located in Xi'an, Chengdu and Shenyang (see Figure 3-1). The parts are then transported and final assembled by COMAC itself in Shanghai. This also means any delay from the Tier-1 airframe component suppliers or in the final assembly will respectively cause hold up in the production rate or accumulation of components from the suppliers. Hence, in order to maximize the overall production rate, it is important that disruptions in each assembly line at different sites will have to be minimized.

Figure 3-1 COMAC ARJ21 Tier-1 Manufacturing and Assembly Network

3.2 The Structure Assembly of ARJ21

The main sequences of the structure assembly processes are depicted in Figure 3-2. At the ARJ21 structure assembly line, each ARJ21 arrives in seven substructures: nose section, front fuselage, central fuselage, aft fuselage, rear fuselage (including tails) and both wings. The components are uploaded to transporters and taken to three specific assembly stations, where in parallel the forward fuselage is constructed of the nose section and front fuselage, the wings are joined to the central fuselage and the aft and rear fuselages are joined which form the aft fuselage (see Figure 3-2). The three main fuselage substructures are then transported to the final assembly station where they are joined together into a complete airframe.

Similar to B747 and B777, ARJ21 uses the method that assemble the middle fuselage and wings together as the middle section first and then assemble it with front section and aft section. The front section and the aft section are also assembled in Shanghai. As Figure 3-2 shown, the front section is assembled from nose and front fuselage, the aft section is assembled from aft fuselage and rear fuselage which includes the vertical stabilizer. The horizontal stabilizer will be assembled later in the system assembly stages (which are carried out after structure assembly processes) as it is an all moving stabilizer. Each sections (i.e. front section, middle section and aft section) are assembled separately and can be done at the same time. The final joints which are numbered as Stage 200A and Stage 200B are also parallel sequences.

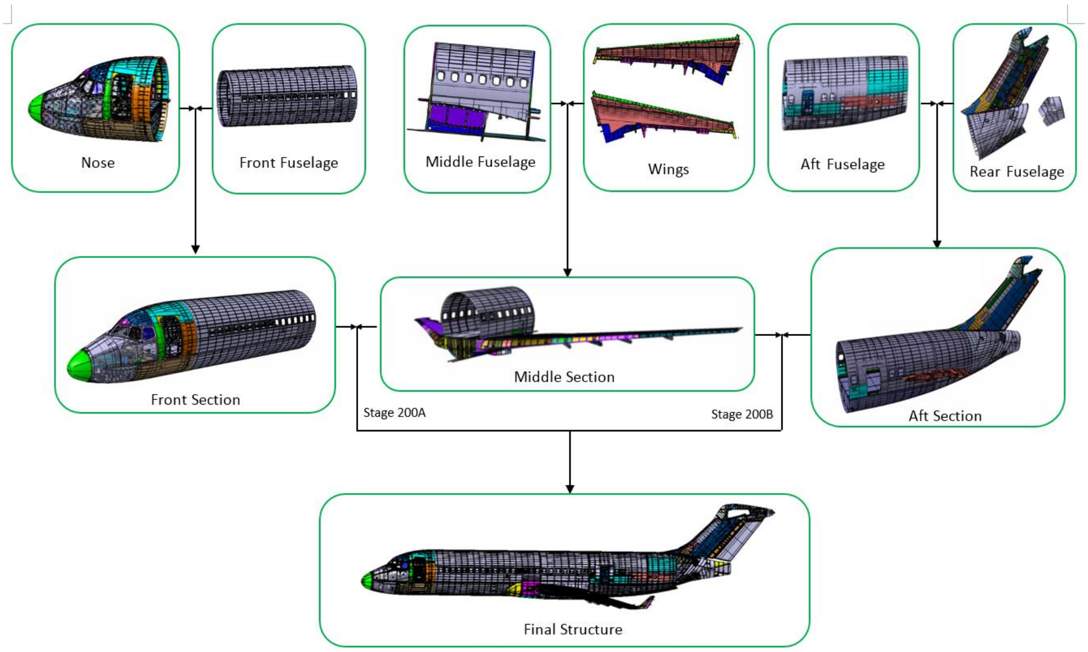


Figure 3-2 Structure Assembly of ARJ21

3.3 Final Assembly Process of ARJ21

In this thesis, the final assembly of ARJ21 fuselage joint (i.e. Stages 200A and 200B) is used as a case study to illustrate the impact of maintenance on the assembly process performance. This can be subsequently extended to cover the whole final assembly process or applied to the other Tier-1 component-level assemblies.

The main sequences of the final joints (Stage 200) can be divided in 5 steps as Figure 3-3 demonstrates. The joint work starts with loading fuselage components into jigs and locating each components roughly. These components are then located accurately by adjusting the jacks according to the measuring data from laser devices. The next operation is to drill joint structures (i.e. skins, frame, stringers and other joint parts) and then bolt the fasteners (i.e. rivets) after deburring. These work are now operated manually but could be done by (semi-

)automatic machines such as “Fuselage Flex Track” (a light weight portable CNC drilling machine) and “Handheld Electromagnetic Rivet Gun” in the near further. The fourth step is about inspection of the riveting quality such as the position, depth and angle of rivets which is usually done manually. And the final step in this stage is unloading the fuselage from the assembly jig.

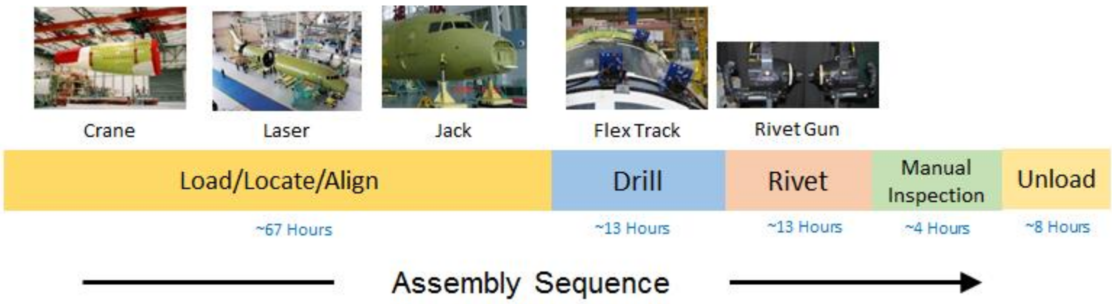


Figure 3-3 Main Sequences of Final Joints and Related Assembly Machines

The five steps in Stage200 take at least 105 hours for each aircraft. Figure 3-3 also depicts the ideal manufacturing hours for each step. But usually it will take more than the ideal hours in this stage because the machines could break down or require maintenance during the manufacturing processes. These data were estimated by COMAC engineer. The detailed steps and lead time of each step are demonstrated in Table 3-1. Step 0210 and step 0220 usually are carried out at the same time and take 13 hours each. The same to step 0310 and 0320.

Table 3-1 Working Processes and Lead Time

Operation Work Contents				Machines	Lead Time (Hrs)
0100	Load/Locate/Adjust				***67***
	0110	Loading			**28**
		0111	Load Middle Fuselage/Wings	Crane	9
		0112	Install Jigs (for laser tracker)		19
		0113	Install Laser Tracker		
		0114	Adjust Jacks		
		0115	Unload Laser Tracker and Jigs		
	0120	Nose/Front Fuselage & Aft Fuselage/Rear Fuselage			**39**
		0121	Load Nose Section/Front Fuselage	Crane	9
		0131	Load Aft Fuselage/Rear Fuselage		9
		0132	Install Jigs (for alignment)		21
		0133	Install Level Instrument		
		0134	Install Laser Target and Laser Gun		
		0135	Adjust Jacks		
		0136	Install Laser Equipment		
		0137	Unload Laser Equipment		
0200	Drill (In parallel)				***13***
	0210	Middle Fuselage/Wings & Nose Section/Front Fuselage			**13**
		0211	Load Flex Track		2
		0212	Drilling	Flex Track	10
		0213	Unload Flex Track		1
	0220	Middle Fuselage/Wings & Aft Fuselage/Rear Fuselage			**13**
		0221	Load Flex Track		2
		0222	Drilling	Flex Track	10

		0223	Unload Flex Track		1
0300	Rivet (In parallel)				***13***
	0310	Middle Fuselage/Wings & Nose/Front Fuselage			**13**
		0311	Load Rivet Gun		2
		0312	Bolting	Rivet Gun	10
		0313	Unload Rivet Gun		1
	0320	Middle Fuselage/Wings & Aft Fuselage/Rear Fuselage			**13**
		0321	Load Rivet Gun		2
		0322	Bolting	Rivet Gun	10
		0323	Unload Rivet Gun		1
0400	Inspection (Manual)				***4***
	0410	Middle Fuselage/Wings & Nose/Front Fuselage			**2**
	0420	Middle Fuselage/Wings & Aft Fuselage/Rear Fuselage			**2**
0500	Unload				***8***
	0510	Install Temporary Landing Gear			**6**
	0520	Unload Final Structure from Gigs			**2**
					105

3.4 Manufacturing Bottlenecks and Assembly Machines

3.4.1 Manufacturing Bottlenecks

In conventional aircraft manufacturing, drilling and bolting are usually completed manually with simple handheld machines such as drillers and pneumatic riveting guns. The processes of drilling and bolting are complex and the quality of products is highly depends on the experience and skills of operators. For example, to drill a row of holes on the skin of fuselage, the operators should first draw the lines and points (with special pens) on the skin manually to locate the holes and then do the drilling work for at least two rounds(get initial holes first and then enlarge them to the final scales). After drilling, deburring must be carried out between each interlayer of materials to avoid the initial fatigue of structure parts. Bolting is not as complex as drilling, but it is still a hard physical work with loud noise to operators.

Manual work may be suitable at the initial stage of the aircraft manufacturing, but limits the production rates and the stability of quality and makes against to the further development of a company. So the application and maintenance of (semi-)automatic machines becomes eagerly required to some companies like COMAC.

3.4.2 Assembly Machines

Nowadays, a variety of automatic assembly machines are widely used in aircraft manufacturing companies. They are being applied in various assembly processes from automated fuselage alignment [28] to robot aided aircraft surface inspections [29]. Considering the actual facts and future planning of COMAC

, the case study of this thesis only focuses on the maintenance of overhead crane, automatic drilling machine and semi-automatic riveting machine as they were identified by COMAC engineers to be subjected to breakdowns and maintenance. The following paragraphs will give a brief introduction of these machines.

Overhead crane is used to carry the components of aircrafts. It could carry component smoothly and steady in three directions. In this case, the overhead crane is only in charge of carrying fuselage to assembly jig. The align work would be done by laser and other related equipment.

Flex Track is a light weight portable, automated, CNC-controlled drilling machine which have been implied in several types of aircraft such as Boeing 767,777,787 and Embraer 170,175,190,195 [30].Manual drilling requires several rounds of drilling with different kinds of drilling bits to reach the final scales while the Flex Track system can consistently and accurately drill and even countersink holes on the skin of an aircraft which drives a leap of the drilling efficiency.

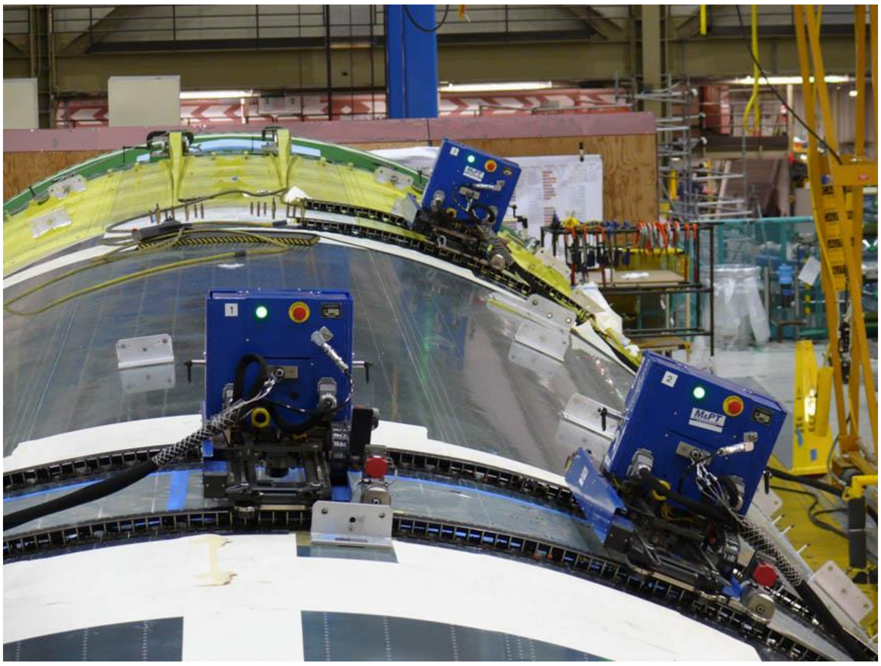


Figure 3-4 Picture of Flex Track Units

Handheld Electromagnetic Rivet Gun is a semi-automatic machine designed to install a variety of solid alloy rivets (e.g. headed rivets and slug rivets) in a safe and efficient manner [30].It is controlled by computer systems and can implement riveting work with stable and high quality without the reliance of operators' skills and experience.

The related parameters of these machines like MTBF (mean time before failure) and maintenance time would be listed later in Section 6.2.

4 MAINTENANCE OF ASSEMBLY MACHINES

4.1 Degradation Process

Machines failures can be divided into two categories, random failures and those as a consequence of degradation. In this thesis, we only consider the degradation failures which maintenance strategies can be applied. A simplified degradation process is illustrated in Figure 4-3. The degradation process can be represented by a stochastic process of increasing wear, hence decreasing in system reliability, finally leading to machine failure. The degradation stages can be modelled using either discrete steps or continuous process in time. The failure occurs when the machine degradation stage reaches a certain reliability level (see Figure 4-3 and Figure 4-4). Maintenances are used to intervene the degradation process and bring about an improvement to a certain reliability level before failures occur. However, when there is no ambiguity, the term ‘maintenance’ will also include ‘repair’ operations in this thesis. The randomness (being stochastic) are from uncertainties in the degradation rate and maintenance.

4.2 Maintenance Strategies

4.2.1 Maintenance Types

The purpose of maintenance is to increase the mean time to failure. It is assumed that maintenance will bring about an improvement to the conditions in the previous stage of degradation [31]. Basically, there are two main types of maintenance strategies named Corrective Maintenance and Preventive Maintenance [32] (see Figure 4-1).The obvious difference between them is whether the maintenance carries out after or before the failure occurrence.

Corrective Maintenance is a retro-active strategy as action is only taken when a system or component failure has occurred and Preventive Maintenance implements before equipment or systems fail (See Figure 4-2). Corrective Maintenance can also be described as "repair" and usually be carried out on items where the consequences of failure or wearing out are not significant and the cost of this maintenance is not greater than preventive maintenance which is not competent for the costly, high-tech and crucial (semi-)automatic machines in aircraft manufacturing industry.

Preventive Maintenance is conducted to keep equipment working and/or extend the life of the equipment [5] which is based on the understanding that a piece of equipment goes through degraded states before failure [11]. It can be divided into two subgroups further, which are Scheduled-based (Pre-Determined) Maintenance and Condition Based Maintenance (CBM).

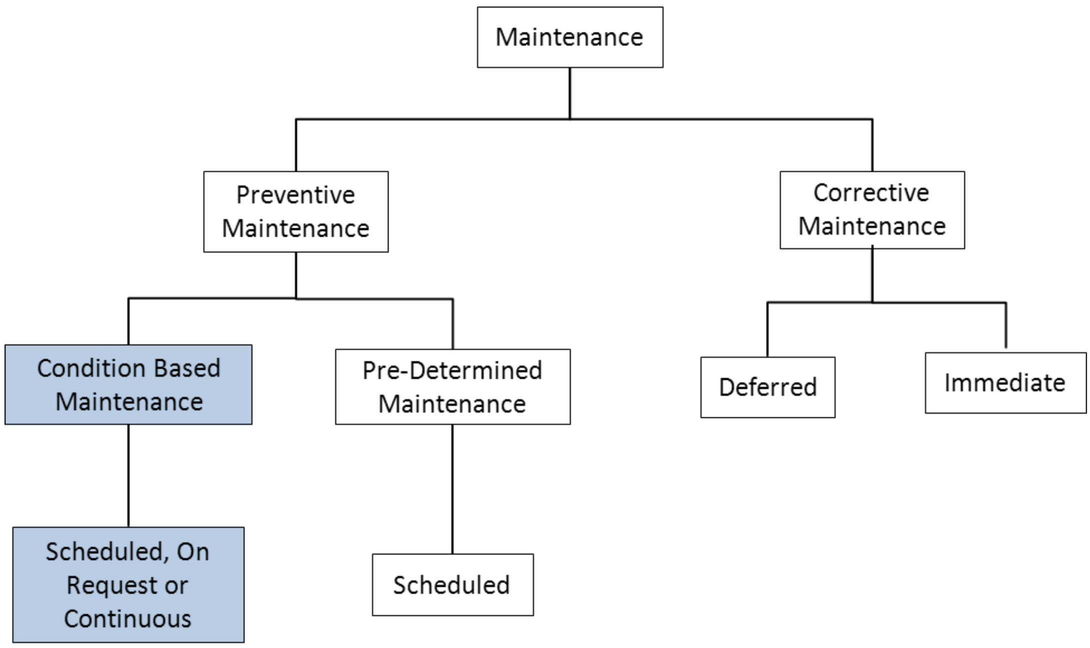


Figure 4-1 Maintenance Types

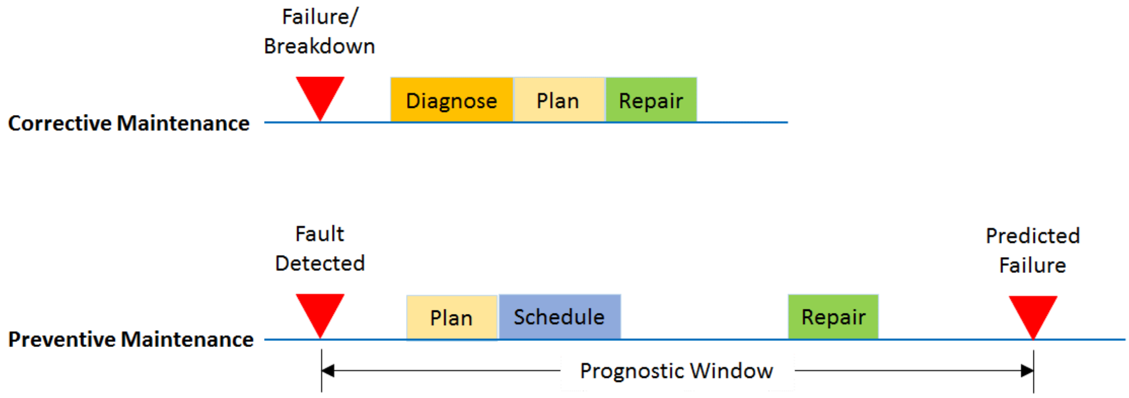


Figure 4-2 Difference between CM and PM

4.2.2 Schedule Based Maintenance

As the name suggests, Schedule-based Maintenance dose the maintenance by schedules and usually at fixed intervals. Figure 4-3 depicts how the Schedule-based maintenance works. The parameter T_S is the scheduled interval of each maintenance and the gaps T_{m1} and T_{m2} are the maintenance time. Most of the time, machines could work well by appropriate Scheduled (Pre-Determined) Maintenance, but sometimes (e.g. the increase of running time of machines due to an extra order from customers) the machines would break down (see T_b) before the next scheduled maintenance (see T_3). The sudden and unexpected breakdowns may require a long period of time to wait for the response of maintenance service and cause a loss of production rates, or cost a big money to shorten the response time.

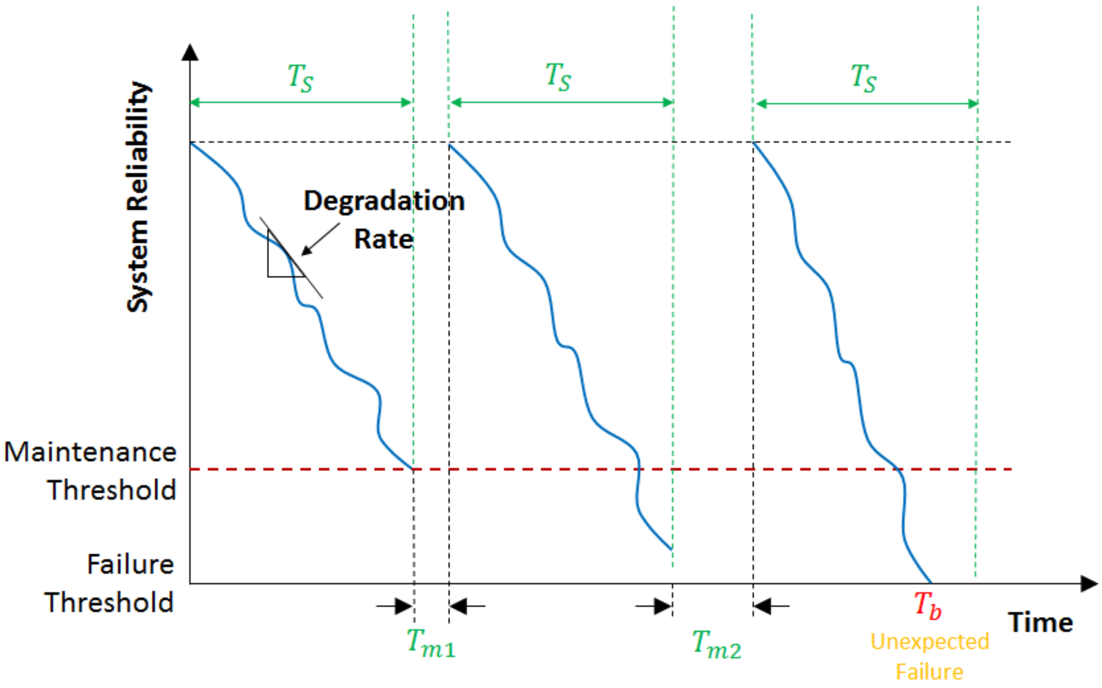


Figure 4-3 Scheduled-Based Maintenance: Machine is maintained based on pre-defined fixed schedule regardless of its condition. Breakdowns caused by degradation can unexpectedly occur before the next scheduled maintenance.

4.2.3 Condition-Based Maintenance

Condition Based Maintenance (CBM) is a methodology with real-time monitoring and performs while the equipment are going to fail or their performances are deteriorating according to the indicator(s).The continuous condition monitoring can be carried out by embedded sensors or periodic inspection as T_i shown in Figure 4-4 (i means the ith monitoring) [14, 11, 33]. This thesis assumes embedded sensors are used to support online continuous condition monitoring (CM). In CBM, instead of traditional fix schedule, maintenance interventions are performed only when the system reliability degrades below a certain preventive maintenance threshold (V^{mtn}). See Figure 4-4, the system reliability is found below V^{mtn} in T_3 and then the maintenance is carried out. Here T_{m1} and $T_{m2}$are the maintenance time and T^{Resp} represents the response time of maintenance service. By monitoring the reliability level, unnecessary maintenance actions and unexpected breakdowns can be reduced.

In CBM, when to take maintenance actions, i.e. defining the maintenance threshold (V^{mtn}), is essentially the main design maintenance parameter. This parameter will be based on both the system reliability level at inspection time and the potential evolution of the system's degradation process. Maintenance threshold must be sufficiently high to allow maintenance actions to be performed before the machine degrades to the failure level.

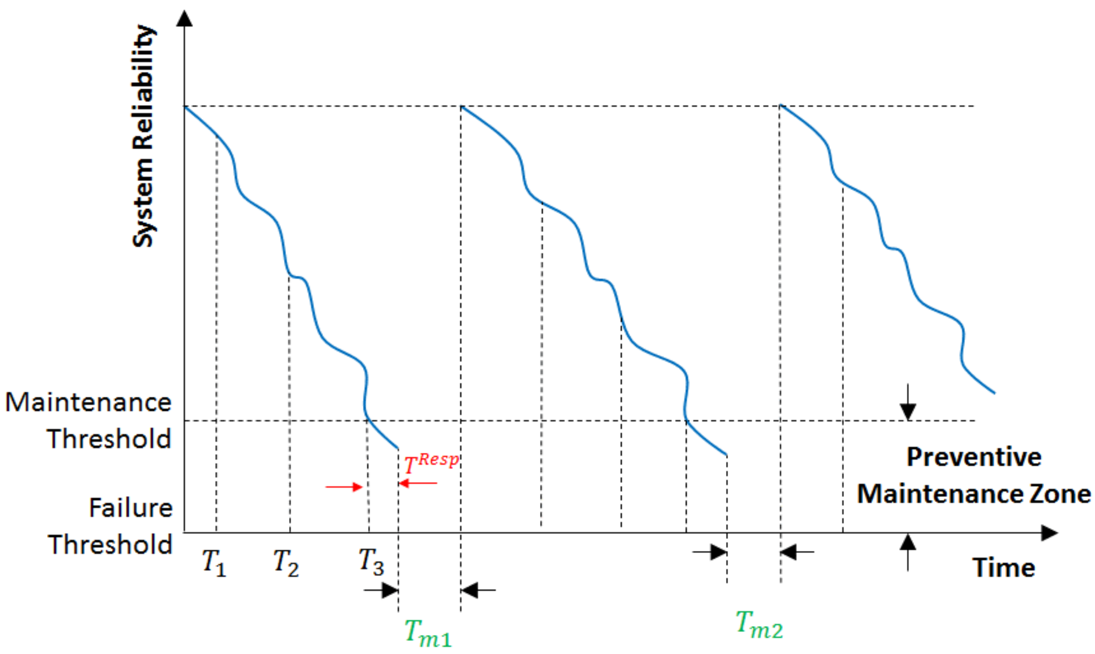


Figure 4-4 Condition-Based Maintenance: With real time monitoring embedded, machine is maintained based on its degradation condition. Cooperating with suitable maintenance threshold and response time, unexpected breakdown and unnecessary maintenance can be reduced.

5 AGENT-BASED SIMULATION MODEL OF ASSEMBLY PROCESS

5.1 Modelling Approaches

5.1.1 Discrete Event Simulation

Discrete Event Simulation (DES) is widely used in modelling and simulation of manufacturing systems [34]. It concerns the modelling of a system as it evolves over time by a representation in which variable states change suddenly at separate points in time and these changes happened in the system are considered events [35]. In DES, each embedded object (entity) acts as a real-world system like component supplement, transfer, and machine- /manual-operating process. Typically DES system are thought of as networks of queues and servers. As the ARJ21 assembly processes could be regarded as sequences of operations, DES was considered at the initial stage of building this simulation model.

To begin with the built of the assembly model, the three sections of component (i.e. The front section, middle section and aft section as Figure 3-2 shown) were simplified as one section and only modelled the running of one machine (three kinds of machines in all) in the whole processes as Figure 5-1 depicted.

DES is suitable for a fixed sequence process where entities (e.g. assembly parts) are move from process to process and this concept is strictly enforced in AnyLogic. But as the development of the model building, DES becomes harder to model more complex interactions between the objects. The most important reason is that the entities in DES are passive, which means something is done to the entities while they move through the system and the intelligence (e.g. decision making) is modelled as part in the system [36, 37].

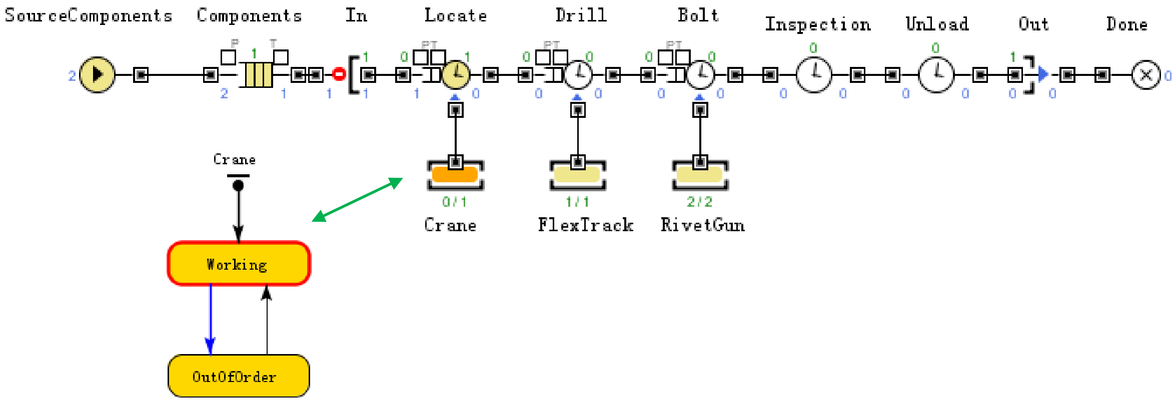


Figure 5-1 Discrete Event Model of Assembly Sequences: Maintenance schedule is modelled in terms of the machine's operable cycle which is controlled by a statechart, see Crane for example. Its interval and action are parameterized by time delays between the state transitions.

In this ARJ21 assembly system, the entities (i.e. machine, maintenance and process) have complex active interactions between each other. The assembly system comprises of machine degradation, maintenance, service operation and assembly process (see **Figure 5-2**). It is a system of multiple processes, and these processes are interacting with each other. Machine degradation has effects on assembly process and maintenance, maintenance has effects on service operation, and service operation has effects on assembly production and cost. Moreover, CBM enable machines will have self-aware behaviour which itself can trigger maintenance operation based its degradation level. Similarly is true for the service operation. To model self-aware and interactions in DES, an overall supervision process has to be created to monitor every single machine's degradation, control service operation on those machine and enforce the maintenance effects on the assembly process. This is counter intuitive and difficult to model as multiple processes have to be tracked and controlled, instead of naturally being autonomous. On the other hand, the three types of machine in this assembly system have very similar activities but different parameter values. Even only model the assembly processes without considering the maintenance, it would still be a little prolix to repeat the similar maintenance and equipment activities three times. Based on these two reasons, especially the first one, DES is not a very advisable choice to this ARJ21 assembly model.

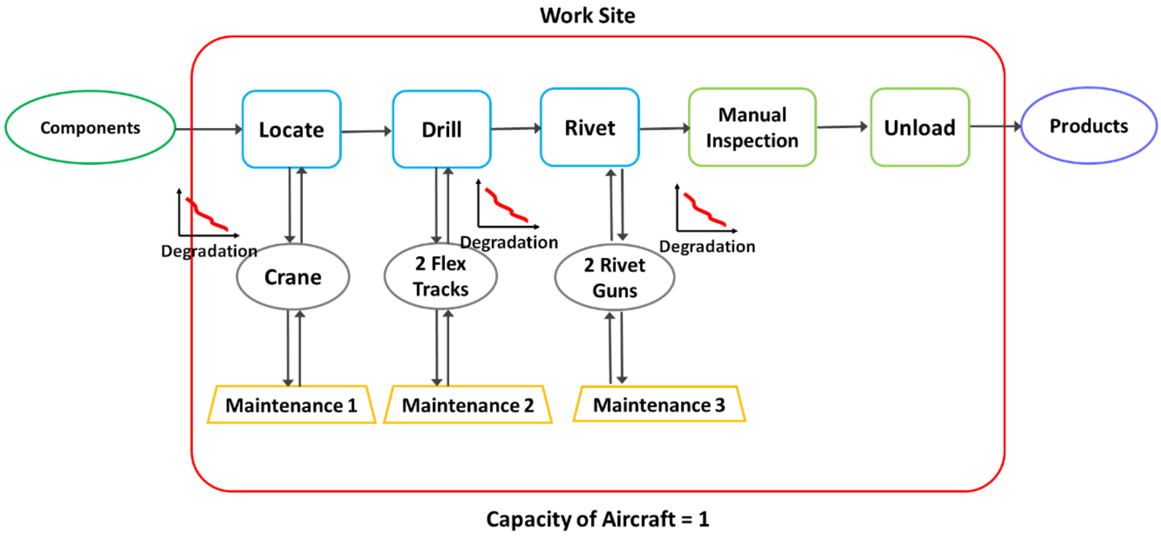


Figure 5-2 Condition Based Maintenance Assembly System: In addition to fix assembly sequence, multiple entities and interactions are needed to capture degradation, self-trigger and maintenance processes. Active self-aware property and multiple interacted entities (in particular for the assembly machines) are needed for modelling a CBM enabled assembly system. These modelling capabilities are not currently supported in the AnyLogic's DES platform.

5.1.2 Agent-Based Simulation

Agent based modelling is much newer than discrete event modelling and can gain deeper insights into the systems traditional modelling approaches do not able to capture well [9]. Different from DES, ABS do not have the concept of queues and flows. Intelligence is represented within each individual entity and the entities themselves can take on the initiative to do something. These entities, which could also be named as 'agents', follow a series of predefined rules to achieve their objectives whilst interacting with each other and their environment [37, 38]. The ‘agent’ in ABS could be multitude of different things, from machines to maintenance engineers.

In this CBM model, five machines of three types are needed in the assembly processes and the maintenance strategies of them are the same but with different parameter values. So each machine could be considered as an agent and this type of agent is named ‘Equipment’ in this model. The main task of the ‘Equipment’ agent is to simulate the state (i.e. in ‘operation’ or ‘out of order’) of machines. Another type of agent in this model is ‘Service’ agent, whose work is to simulate the state (i.e. ‘idle’, in the process of ‘response’ or doing ‘maintenance’) of maintenance engineers.

5.2 Agent-Based Simulation Modelling

5.2.1 Model of Assembly Machines

Figure 5-3 illustrates the statechart of “Equipment Agent” and how the statechart works. The main state of the agent are “Operation” and “Out of Order” which are triggered by running out of “Remaining Useful Life” (RUL<=0) or the demands of maintenances (Maintenance Requirement Responded). After maintenance, message “Finish” which is send by “Service Agent” would be received and then the machine turn back to “Operation”.

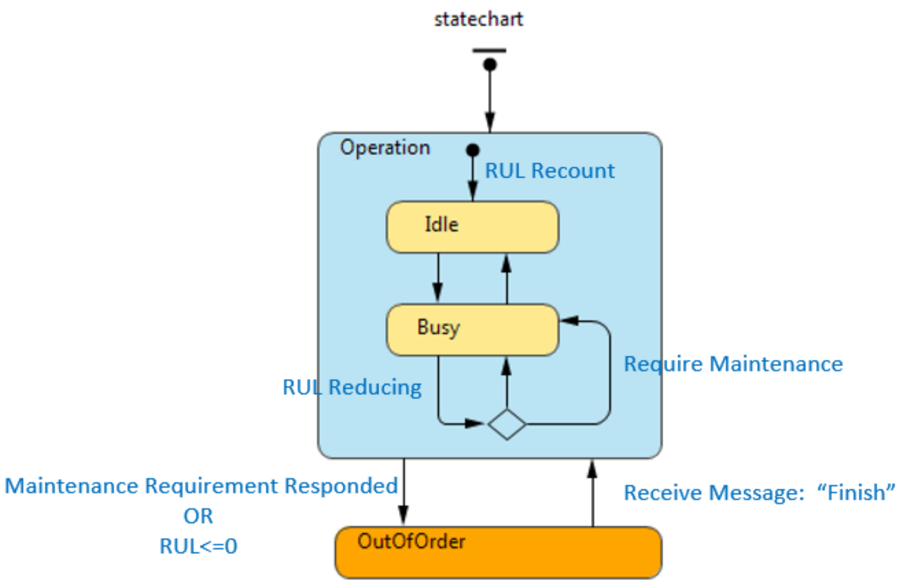


Figure 5-3 Equipment Agent Model

Inside “Operation” state, RUL time reduces while the machine is in the “Busy” state and do not change in the “Idle” state. When RUL is lower than the maintenance threshold, the machine sends a message “Require Maintenance” to “Service Agent” to require maintenance.

5.2.2 Model of Maintenance

Figure 5-4 demonstrates the “Service Agent” whose objects are the service engineers. While the engineers receive the message from “Equipment Agent” to do the maintenance, they switch their states from “Idle” to answering the “Response” of requirements, the response time is one of key parameters in this thesis which would be mentioned later in Section 6. After the engineers arrive, they will spend “Mean Maintenance Time” (MMT) in maintaining the machine and then back to the state of “Idle” if there is no other machines need to maintain.

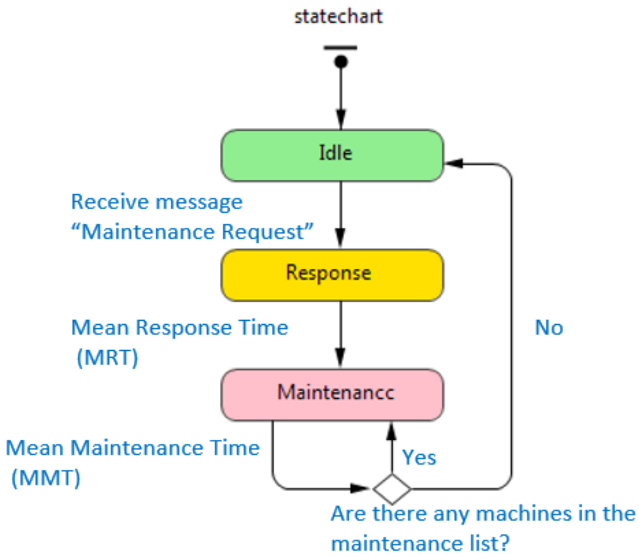


Figure 5-4 Service Agent Model

5.2.3 Model of Assembly Process

The main part of this model can be simplified as the statechart shown in Figure 5-5. Each block (from locating process to unloading fuselage and back to locating) stands for the assembly processes of one aircraft and takes 105 simulated hours according to Table 3-1.

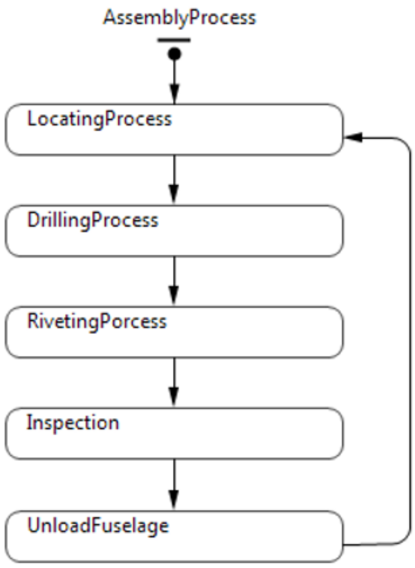


Figure 5-5 Assembly Processes Chart

The processes that need to use (semi-)auto machines are the first three steps: locating process, drilling process and riveting process. The other two steps will take some time to pass through the flow but do not have relationship with the agents.

As there is only one crane but three components in this model, the crane need to transfer components one by one. The processes are putting one component to the jig first and then check whether all the three components are on the jig. If the jig is still available (have room for new components), the crane keeps on transferring the remaining components until the jig is fully occupied with all three components. After this process finished, the components go to the next state (i.e. drilling process) and the crane go back to 'Idle' and wait until the next fuselage to be assembled (See Figure 5-6). The processes of drilling and riveting are similar. Loading the equipment and then doing the work until operation time is equal or bigger than the required operation time as Figure 5-7 shown.

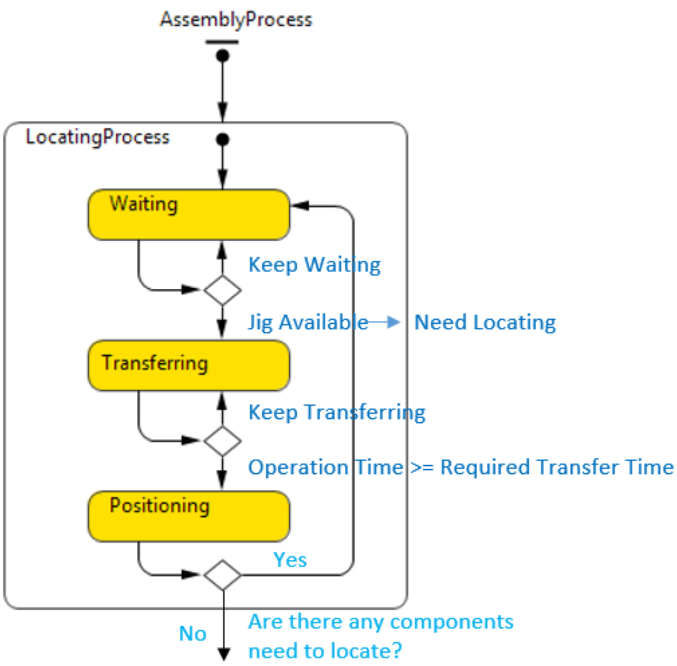


Figure 5-6 Locating Process Chart

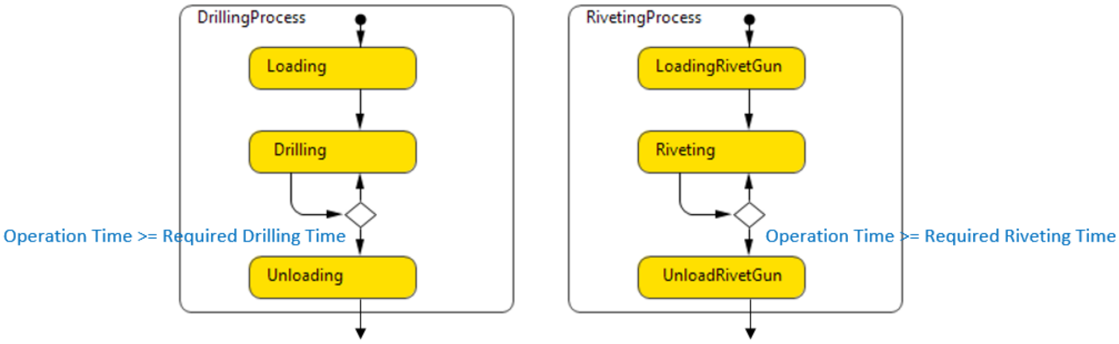


Figure 5-7 Drilling and Riveting Process Charts

5.2.4 Interaction between Agents

Figure 5-8 depicts the interaction between agents. Taking “Crane” as an example, for each single machine, the initial state is “Idle” and turn to “Busy” after receiving the message “Work in Operation” (see Figure 5-3). The message is given by the main assembly processes when the sequences switching from “Waiting” to “Transferring” (see Figure 5-6). After transferring, the chart goes to “Positioning” and sends a message “Work Completed” to the “Equipment Agent” to let the machine turn back to “Idle”.

During the operation, RUL of the machine keep reducing until it is out of order. There are two situations that the machine is out of order. The first one is running off RUL and breaking down itself, the other one is doing the maintenance and need to switch off the machine. As this thesis mentioned before, the first situation is the one that usually company do not want to meet with as it could cause a delay of manufacturing and reduce the product rate. In the second situation, the flow goes into the “Service Agent” when RUL is equal or less than the threshold value as Figure 5-8 shown in the yellow rectangle. Threshold here is another important parameter in these thesis. While the machine is triggered in the “Equipment Agent” (see Figure 5-3), a message “Maintenance Request” is sent to the

“Service Agent”. After the response time, RUL stop reducing (if RUL value is still greater or equal to zero) and the machine turn to “Out of Order”. After maintaining, the machine goes back to the “Idle” again.

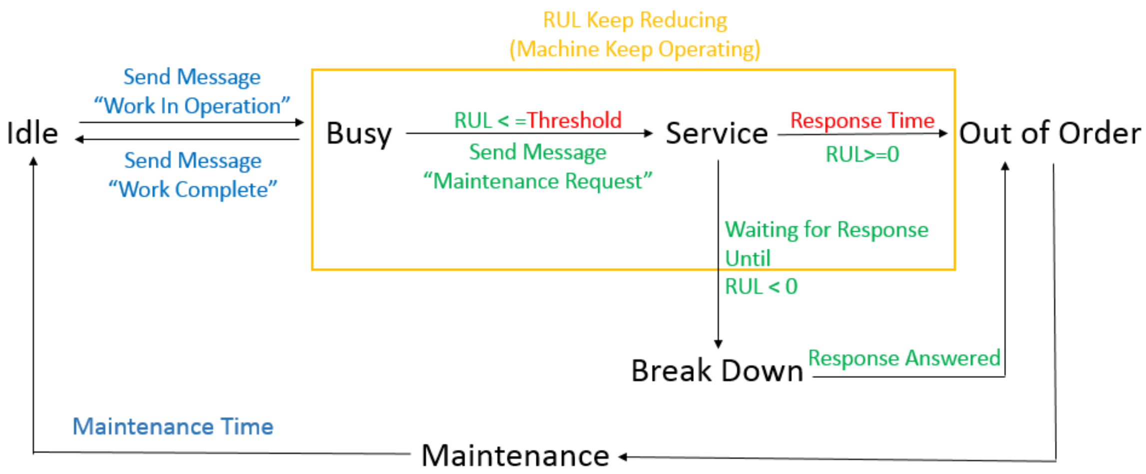


Figure 5-8 Interaction between Agents

A little different from crane, the drilling machine and riveting machine each of them has only one service but two sets of machine (see Figure 5-9).In this case, the model need to judge whether the service is busy or not and these two machines (could be much more than two) will be maintained one by one like a queue.

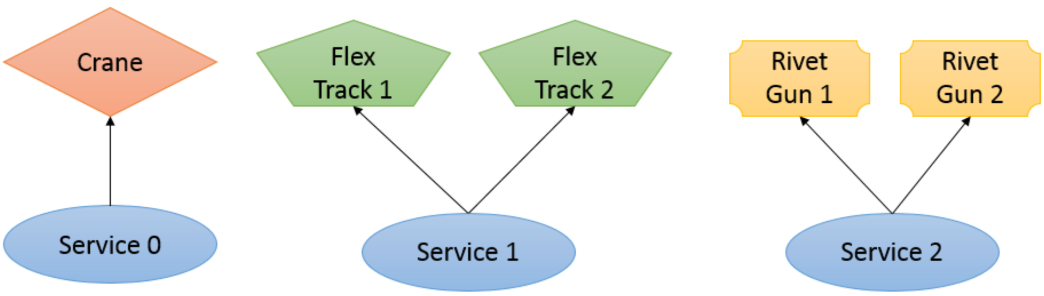


Figure 5-9 Relationship between Equipment and Service

6 MULTI-OBJECTIVE OPTIMIZATION

6.1 Simulation Optimization

A simulation model describes the input-output behaviour of a complex system which are transformed from the real-world problem and works as a function (whose explicit form is unknown) [39, 40, 41]. The input parameters here are maintenance threshold of machines and maintenance response time of services as Figure 6-1 shown. After simulating the assembly processes with uncertainties and constraints (as the red rectangle ‘simulation model’ shown in Figure 6-1), a set of output (which is ‘production loss’ and ‘maintenance cost’ in this case) could be obtained. Because of the existence of uncertainty, multiple replications will be used to get the mean values which would be discussed later in section 6.3.2. For decision makers, obtaining the simulation results is far less than enough. Getting a trade-off between objectives to find the optimal combination of conditions resulting in the possible solution is the main aim of simulation optimization.

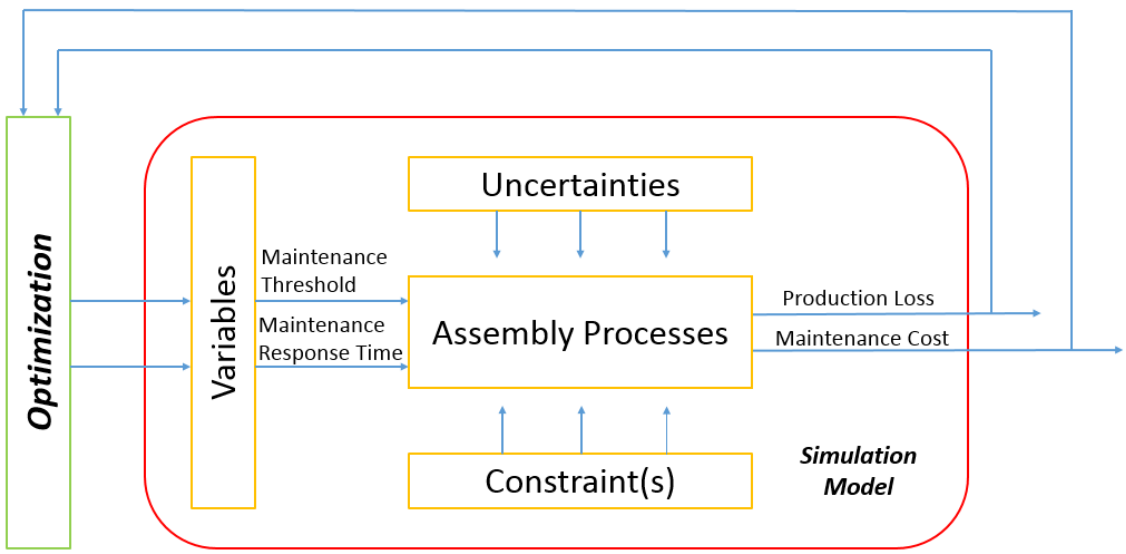


Figure 6-1 Interaction between Simulation Model and Optimization Module

Optimization is a process of finding an optimal combination of conditions resulting in the best possible solution [42]. It is an iterative process which requires a number of iterations of simulation to reach the optimal value. See Figure 6-1, the optimization engine gets the data from simulation model and gives feedback to the simulation model after one integration of optimization process. The integration processes will keep on running until the optimal solution is found or reaches the predetermined number of interaction. As the simulation model do not provide the capability of finding the optimum set of decision variables, an optimization engine would be essential. The OptQuest Engine, which is embedded in AnyLogic, provides a tool to calculate optimal solutions for the decision variables. It uses metaheuristics to guide its search algorithm toward better solution which would be discussed later in section 6.3.

In brief, the simulation model is a kind of function with inputs and outputs and the optimization engine works as a hunter to find an input with the optimal output in an iterative process.

6.2 Objective Functions

6.2.1 Production Loss

In this case, it is supposed that factors like breakdown of supplement chain, mood of workers, nature disasters do not have any effect on the results of this model. The only factor need to be considered in this case is the condition of machines, which means while the manufacturing processes keep on operating without any stops to maintain any machines, this system can get the maximum production rates. As Table 3-1 shown, it takes 105 hours to produce one aircraft ($T^{preAirC} = 105$). To make the results more distinct, we expand the simulation time period from one year to ten years ($T^{prodPeriod} = 10years$). So the maximum number of the aircraft (N^{max}) which could be produced in ten years is about 834.286 as (6-1) shown.

$$N^{max} = \frac{T^{prodPeriod}}{T^{preAirC}} = \frac{10years \times 365days \times 24hours}{105hours} \quad \textbf{(6-1)}$$

Because of the unavoidable maintenances and breakdowns of machines in the processes of production, the real output of aircrafts will be less than the maximum value. In this model, we define a parameter $R^{prodLoss}$ to measure the loss of production:

$$R^{prodLoss} = \frac{N^{max} - N^{Aircraft}}{R^{AcPL} \times N^{max}} \quad \textbf{(6-2)}$$

Here $N^{Aircraft}$ represents the number of aircrafts be produced in the simulation period and $R^{AcPL} = 5\%$ stands for the acceptable production loss which works as a normalizing constant in this equation.

For multi-objective optimization, the crucial data is the relationships between input and output but not the real numbers of them. To make the results clear and easy to be understood, the parameters could be multiplied by any positive numbers. That is why here could have a normalizing constant.

6.2.2 Maintenance Cost

Maintenance cost in this case contains two parts. One is named fixed cost ($FixCost$) and the other is response cost ($ResCost$).Fixed cost is the part that be accounted by the number of maintenance preformed and each time costs the same price for a type of machine. The response cost depends on the response time which means the price increases with the reducing of the response time. Different machines have different fixed cost and response cost. Parameters $W^{FixCost}$ and $W^{ResCost}$ in Equation (6-3) present the weights of them. Here $W^{ResCost}$ bases on the standard Mean Response Time (Thus, for each maintenance, the cost (C^{preMtn}) would be

$$C^{preMtn} = W^{FixCost} + W^{ResCost} \times \frac{\mu^{T^{Resp}}}{T^{Resp}} \tag{6-3}$$

where variable T^{Resp} is the response time of one maintenance and $\mu^{T^{Resp}}$ represents the Mean Response Time as Table 6-1 shown.

Table 6-1 Maintenance Cost Parameters

	$W^{FixCost}$	$W^{ResCost}$	$\mu^{T^{Resp}}$ $(hour)$	$N_c^{mtn}(times)$
Crane	1	1	168 (1 weeks)	31.3
Flex Track	2	1.5	504 (3 weeks)	18.5
River Gun	0.4	0.6	336 (2 weeks)	11.6

As there are three different types of machine in the simulation model, the total maintenance cost C^{mtn} in this model would be

$$C^{mtn} = \sum_{i=0}^{2} C_i^{mtn} = \sum_{i=0}^{2} \frac{N_i^{mtn} \times C_i^{preMtn}}{C_c^{mtn}}$$
$$= \sum_{i=0}^{2} \frac{N_i^{mtn} \times \left(W_i^{FixCost} + W_i^{ResCost} \times \frac{\mu^{T^{Resp}}}{T_i^{Resp}} \right)}{C_c^{mtn}} \tag{6-4}$$

where N_i^{mtn} represents the number of maintenance times and C_c^{mtn} is a normalizing constant. Subscript i stands for the serial number of machine. Number 0, 1 and 2 successively represents crane, flex track and rivet gun.

Normalizing constant C_c^{mtn} here could be treated as a kind of standard maintenance cost as well. It could be a set with Mean Response Time (MRT) which means one standard price go with one MRT value. While the response time T_i^{Resp} is shorter/longer than $\mu^{T^{Resp}}$, the price would be higher/lower than the standard cost. The definition of C_c^{mtn} is based on the data of Table 6-1 as well. Constant N_c^{mtn} means the standard number of maintenance which go with the standard cost. Here the values of N_c^{mtn} come from the minimal maintenance times the machines could have. Thus, this normalizing constant C_c^{mtn} is

$$C_c^{mtn} = \sum_{i=0}^{2} {N_c^{mtn}}_i \times \left(W_i^{FixCost} + W_i^{ResCost}\right) \tag{6-5}$$

$$= 31.3 \times (1 + 1) + 18.5 \times (2 + 1.5) + 11.6 \times (0.4 + 0.6)$$

6.2.3 Design Parameters

There are two types of parameter in this optimization model, the uncontrollable parameters and the controllable parameters. The uncontrolled parameters come from the data of machines and the production sequences as Table 6-2 shown. The “Operation Hours” is based on Table 3-1 which means the total operation hour one type of machine need to product one aircraft in this final assembly process. The controllable parameters are maintenance threshold V^{mtn} and response time T^{Resp} which are the objects of the optimization and work as decision variables in this model.

Table 6-2 Optimization Parameters

	MTBM(Hour)	MMT(Hour)	Operation Hours
Crane	720	1	27
Flex Track	900	2	20
River Gun	1440	1	20

Except these parameters, another important and special variable in this model is $uncertainty$. The meaning of it would be discussed in detail in section 6.3.1. In this model, we define

$$uncertainty = 25\% \tag{6-6}$$

The uncertainty in this model follows the triangular distribution which is primarily used in the duration of operations like service time. Equation (6-6) means the lower and upper uncertainty bounds are assumed at $\pm 25\%$ of the mean values as Figure 6-2 shown. In AnyLogic, the built-in 'triangular' function can be used to generate random samples, where the lower bound, median and upper bound are the function's parameters.

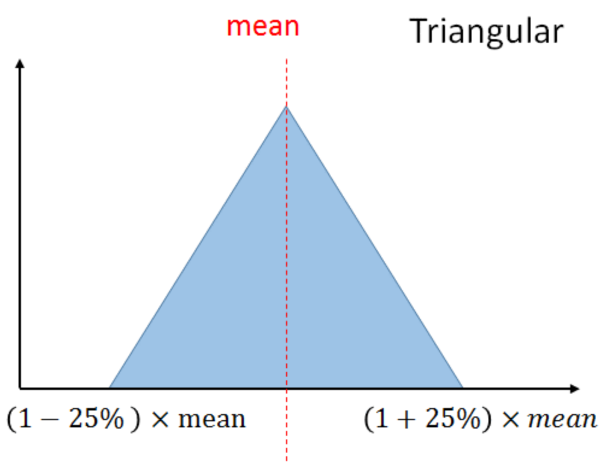


Figure 6-2 Triangular Distribution

There are also other types of uncertainty distributions like the uniform distribution and normal distribution. The uniform distribution has equal probability between minimal and maximal value. The normal distribution is unbounded on both sides which means the value could be lower than zero or infinite though the chance is very small. In assembly process, the operation time (i.e. Lead time of processes and maintenance time of machines) is in general bounded and the value usually has more probability to be around the mean value. So comparing with them, the triangular one is more suitable in this case. The value of uncertainty in this case is based on the experiential estimation from the engineer in COMAC. It could be set as any other value if necessary.

6.3 Uncertainties and Stochastic Optimization

6.3.1 Uncertainties

An optimization approach might be much simpler if there is no uncertainty and randomness in the systems (e.g. deterministic optimization problems). However, many real-world optimization problems involve some sort of uncertainties in the form of randomness like the case in this thesis. This kind of optimization is called "Stochastic Optimization". Because of their complexity and stochastic relations, these models usually are quite challenging to deal with. And also that is the reason why simulation is needed [43, 44, 45]. So the uncertainties is an important and essential concept in this thesis.

In this model, the uncertainties roughly come from three aspects. The first one is the lead time of working processes (see Table 3-1) like transferring time, drilling time and so on. This part of uncertainties are accompanied with the whole of working. They could be caused by the performance of machines or the operations of workers such as loading and unloading equipment. The second part of the uncertainties come from the designed parameters of machines. They include the MTBM (Mean Time before Maintenance) and the degradation rate of RUL (Remaining Useful Life).The last part of uncertainties happen in the processes of waiting and doing the maintenance. The related variables are maintenance response time (T^{Resp}) and MMT (Mean Maintenance Time).Usually the three types of uncertainties should have different valves, but the valves in this case not really affect the trade-off between the objectives, so in the simulation model, they have the same value as (6-6) shown.

6.3.2 Stochastic Optimization

Because of the existence of uncertainties, each round of simulation should have different results except coincidence, though usually the difference is very small. If the iterative optimization was based on only one time of simulation, the optimal result could be deflected from the real one because of the randomness. So for each iteration of stochastic simulation optimization, the stochastic simulator does multiple sampling to obtain the mean value of outputs as the basis of optimization.

Stochastic optimization nowadays plays a significant role in the design, analysis, and operation of modern systems. Methods for stochastic optimization provide a means of coping with process uncertainties. [43, 46] In this case, for example, each round of simulation could obtain a set of output: the production loss ($R^{prodLoss}$) and maintenance cost (C^{mtn}). After N rounds of simulation, we could get a set of mean value,

$$\begin{cases} \bar{R}^{prodLoss} = \dfrac{1}{N}\sum_{i=1}^{N} R_i^{prodLoss} \\ \bar{C}^{mtn} = \dfrac{1}{N}\sum_{i=1}^{N} C_i^{mtn} \end{cases} \tag{6-7}$$

which is the result of this round (one iteration) of simulation optimization and the reference of the next round. Figure 6-3 depicts the structure of a stochastic optimization process where V^{mtn} stands for the threshold of maintenance. The optimal value would finally be found by running multiple iterations of optimization and each iteration contains multiple replications of simulation whose aim is to obtain the mean value of each iteration.

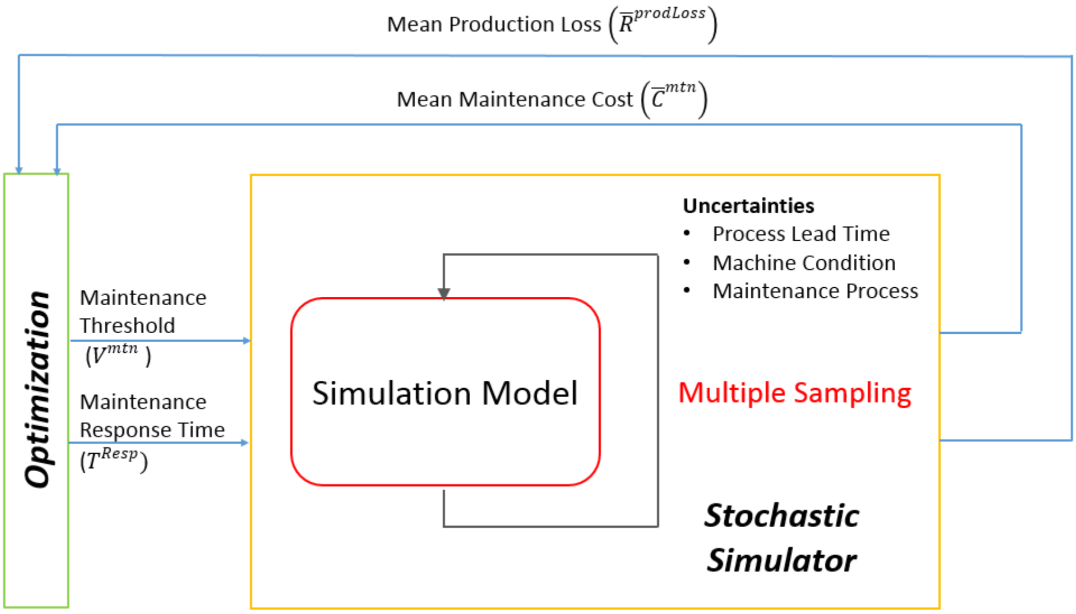


Figure 6-3 Structure of a Stochastic Optimization Process

6.4 Scatter Search

OptQuest Engine is an optimization tool which allows analysts to search for optimal solutions to complex business and engineering problems. It could be embedded in some commercial software like AnyLogic and works as a module.

The OptQuest Engine incorporates metaheuristics to guide its search algorithm towards better solutions. This approach remembers which solutions worked well and recombines them into new, better solutions. Metaheuristics is a family of optimization approaches that includes scatter search, genetic algorithms, simulated annealing, Tabu search, and their hybrids [47, 48]. But for AnyLogic, OptQuest's scatter search algorithm is the only optimization engine/method that be built-in [49] .

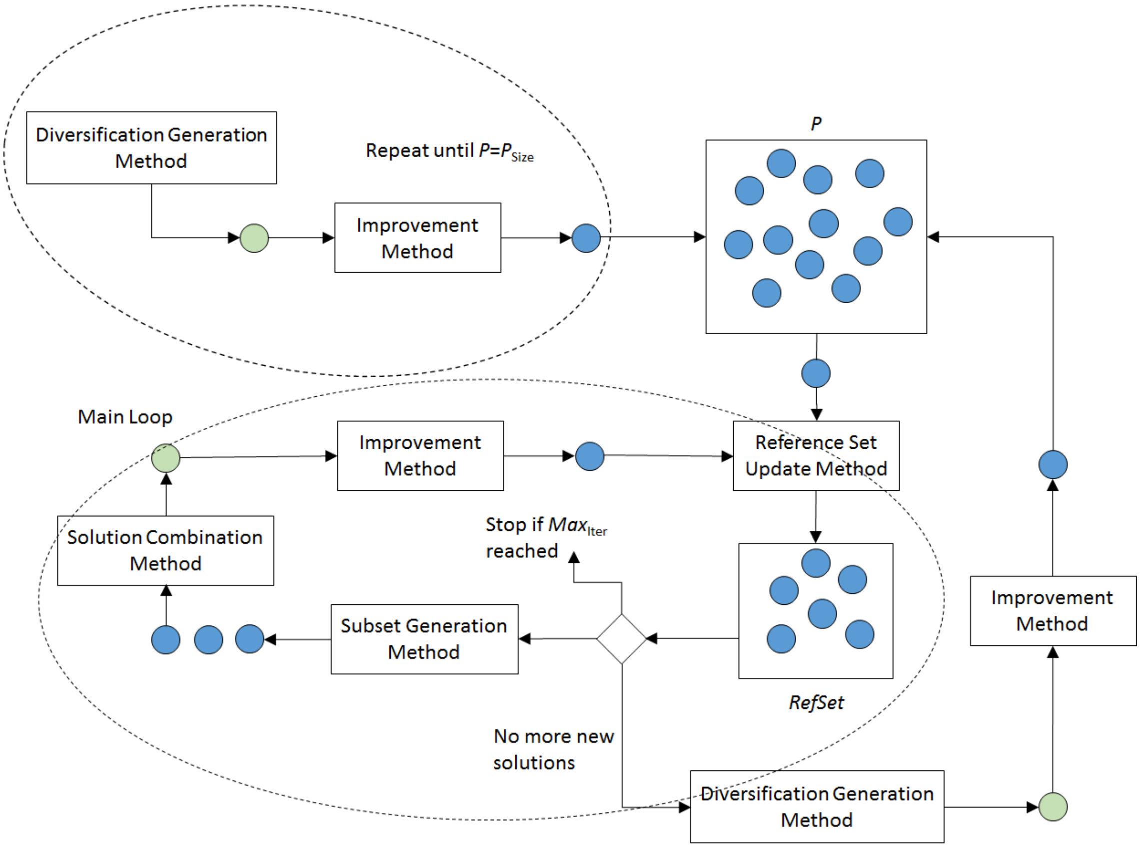


Figure 6-4 Schematic Representation of Scatter Search Design

Scatter search is a population-based metaheuristic for optimization and has been successfully applied to hard optimization problems with continuous and discrete variables [50, 26]. It consists of five methods:

1. Diversification Generation
2. Improvement
3. Reference Set Update
4. Subset Generation
5. Solution Combination

Figure 6-4 illustrates the interaction among these five methods. The design starts with the diversification generation method which is used to generate a large set P of diverse solutions that are the basis for initializing the research. The improvement method transforms solutions with the goal of improving quality or feasibility and the process repeats until $|P| = P_{Size}$ .The initial $RefSet$ is built according to the reference set update method, which can take the b best solutions (as regards their quality or diversity in the problem solving) from P to compose the $RefSet$. The search is then initiates by applying the subset generation method which produces subsets of reference solutions as the input to the combination method. The solution combination method uses these subsets to create new combined solution vectors. Improvement method is used again to get enhanced solutions. The reference set update method is applied once more to build the new $RefSet$ and the main loop repeats again. The repeating stops while no more better solutions appear or the maximum iteration is reached. [49, 51, 52, 53, 54]

6.5 Optimization Approach

6.5.1 Pareto-optimal Solutions

However, it is important to note that the feasible objective space not only contains Pareto-optimal solutions, but also solutions that are not optimal [55]. See Figure 6-5, the aim here is to minimize both cost and production loss. There are seven points in the figure and each point represents a solution. It could be obviously discovered that solution ‘g’ is worse than the other solutions in all objectives. For solution ‘d’ and ‘f’, they have the same production loss but solution ‘f’ would cost more than ‘d’, so solution ‘d’ is better than ‘f’ in one objective. The same to solution ‘b’ and ‘e’. But for solution ‘a’, ‘b’, ‘c’, and ‘d’, they could not be compared in this way as they are all the optimal solutions but have different weight for two objectives.

Thus, these solutions could be classified into two non-overlapping regions, namely one which is optimal and one which is non-optimal. The goals in a multi-objective optimization are:

1. To find a set of solutions as close as possible to the Pareto-optimal front.
2. To find a set of solutions as diverse as possible.

The first goal is mandatory in any optimization task. The second one is entirely specific to multi-objective optimization to have a good set of trade-off solutions among objectives [55, 12].

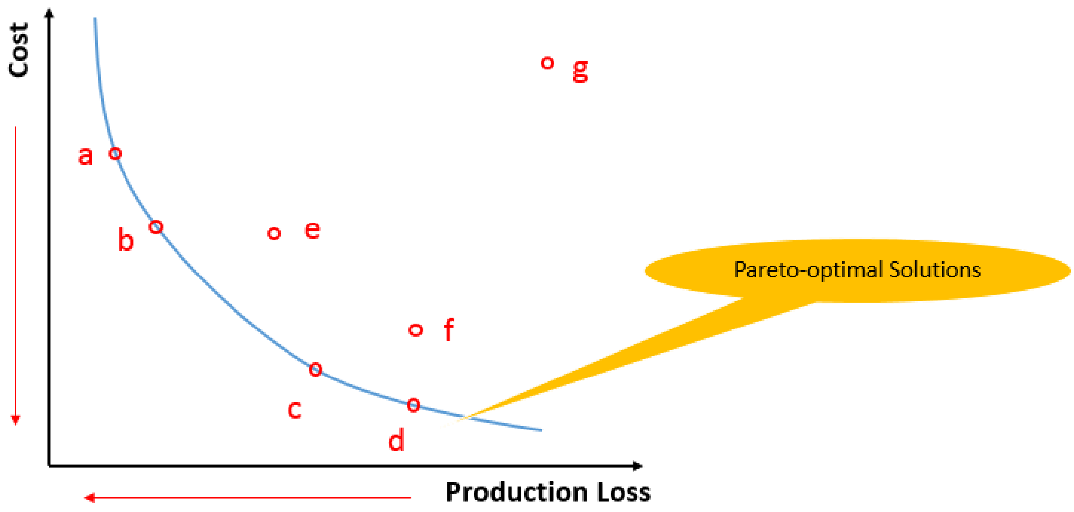


Figure 6-5 Trade-off between Objectives

6.5.2 Multi-objective Optimization

In a single-objective optimization problem, the task is to find one solution (which optimizes the sole objective function [55]. But nowadays, optimization problems usually involve multiple objectives. For example, the case in this thesis, the objectives are production loss ($R^{prodLoss}$) and maintenance cost (C^{mtn}). In general, companies aim to spend the least money to get the highest production rates, which means to get lowest $R^{prodLoss}$ value and C^{mtn} at the same time. But the fact is, while reducing the maintenance response time (T^{Resp}), the operation time of machines could be increased and the $R^{prodLoss}$ value would be reduced. However, the shorter the response time is, the higher value C^{mtn} will be.

From this example, it could be found out that in a multi-objective optimization problem, each objective could have different or maybe sufficient different optimal solution, which means no one can be considered to be better than any other with respect to all other objective functions. It also means, the aim of multi-objective optimization is not to simply find a single optimal solution corresponding to each objectives function. However, there usually exist a set of solutions for the multiple-objective cases which is called trade-off solutions (also named Pareto optimal solutions or nondominated solutions) as Figure 6-5 shown. No improvement in any objective function is possible without sacrificing at least one of the other objective functions [56, 25]. In other words, reducing the production lost usually is at the expense of the increasing of maintenance cost and vice versa. In different period of time, a company may have different requirement. Sometimes the company focuses more on saving cost, other time maybe concentrates more on the production rates. Therefore, the Pareto-optimal solutions could help decision makers to balance the cost and the production loss.

6.5.3 Weighted Metric Methods

6.5.3.1 Distance Metrics

By using OptQuest to deal with the multi-objective optimization problems, there are a few classical methods could be applied, like weighted approach, goal-oriented optimization, and frontier search [57]. Though these approaches are different from each other, the main aim of them is the same, which is to convert a multi-objective optimization problem into a single-objective optimization problem [55].

Among these approaches, the weighted approach is the simplest and probably the most widely used classical approach. It works well when the objectives are well behaved and trade-offs between the objectives allow the weights to be easily determined ahead of time, which is quite suitable for the case in this thesis [57].

The main idea of the weighted approach is pre-multiplying each objective with a user-supplied weight. The weight of an objective is usually chosen in proportion to the objective's relative importance in the problem. For an optimization problem with two objectives, knowing any one, the other weight can be calculated by simply subtraction.

Weighted Metric Methods can be regarded as a kind of mathematical formulation of goal-seeking behaviour in terms of a distance function. For non-negative weights, the weighted l_p distance measure of any solution x from the ideal solution z^* can be minimized as follows:

$$Minimize \quad l_p(x) = \left(\sum_{m=1}^{M} w_m |f_m(x) - z_m^*|^p\right)^{1/p} \tag{6-8}$$

where x the input of optimization model is, $f(x)$ is the output of the model, w means weight and m is the sequence number of the M objectives [55, 56].

Here we assume $m = 2$ and $z^* = 0$, which means there are two objectives and the ideal solutions of them are both zero. While $p = 1$, the method could be regarded as the weighted sum method which is widely used nowadays and the function can be described as

$$Minimize \quad l_1(x) = w_1 f_1(x) + w_2 f_2(x) \tag{6-9}$$

This equation could also be simplified as

$$Minimize \quad L_1 = w_1 O_1 + w_2 O_2 \tag{6-10}$$

With a certain set of weight, equation (6-10) could be regarded as a set of lines with the same slope (which equals $-w_1/w_2$) as Figure 6-6 shown. The location of the line depends on the value of L_1. The aim here is to get the minimal value of L_1 in the search space (feasible objective space) and this happens when the line is tangential to the search space in the bottom-left corner of the space as $line\ a$ shown in Figure 6-6. The tangent point P is one of the points in the Pareto optimal front. By changing the weight w_1 and w_2 ,and getting sets of contour lines with different slopes, more tangent points would be found and a rough Pareto optimal front could be emerged. The more weights be calculated, the closer to the real Pareto optima front.

Similarly, while $p = 2$, the function could be written as

$$Minimize \quad L_2 = \sqrt{(w_1O_1)^2 + (w_2O_2)^2} \tag{6-11}$$

The task here is to find the tangent points of the ovals (with different weights) and the feasible objective space where L_2 get the minimal values as Figure 6-7 shown. For p=1 and p=2, the methods are also known as Norm1 and Norm2 respectively.

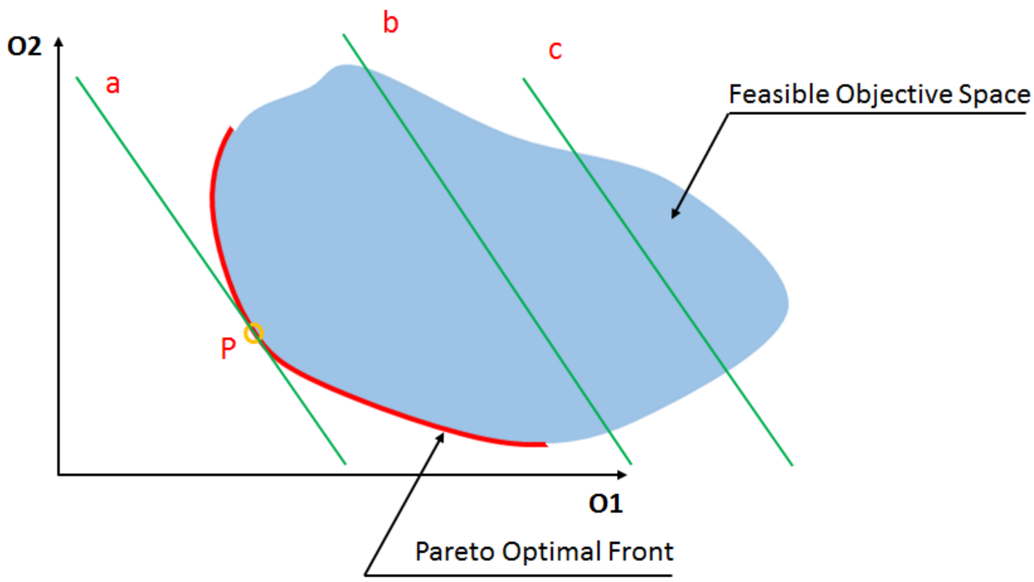


Figure 6-6 the Weighted Metric Method with $p = 1$

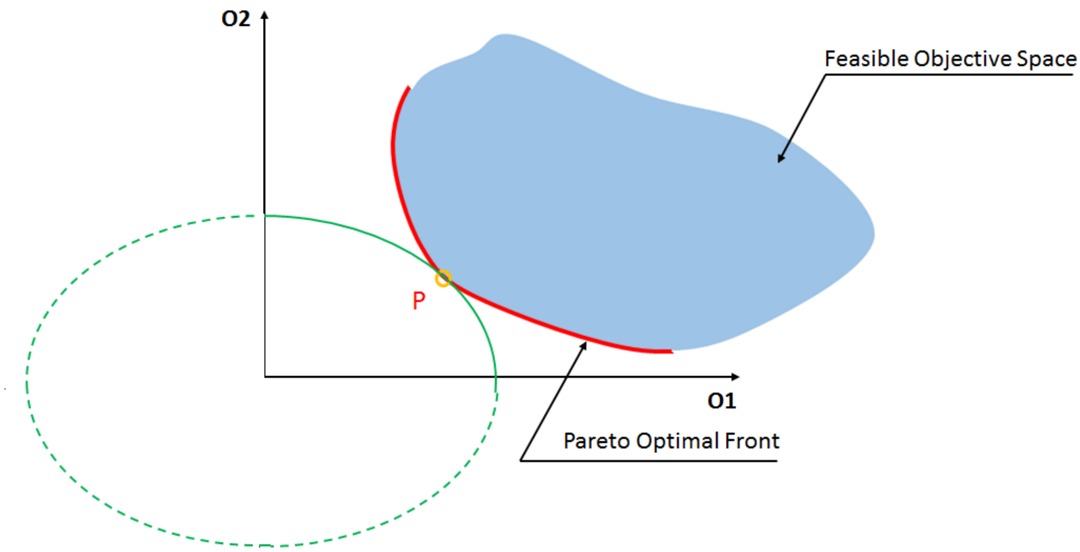


Figure 6-7 the Weighted Metric Method with $p = 2$

While $p = \infty$, the problem becomes a weighted min-max problem [58, 59] and the equation is

$$Minimize \quad L_\infty = max_{m=1}^{M}(w_m O_m) \tag{6-12}$$

For a problem with two objectives, L_∞ are sets of rectangle whose diagonal lines are on the same line as $line\ s$ shown in Figure 6-8. The slope of it is w_1/w_2. Comparing with weighted sum method ($p = 1$), weighted min-max method ($p = \infty$) is more visualized (this part will be discussed further more in the next subsection). In this thesis, weighted min-max method is applied in optimization.

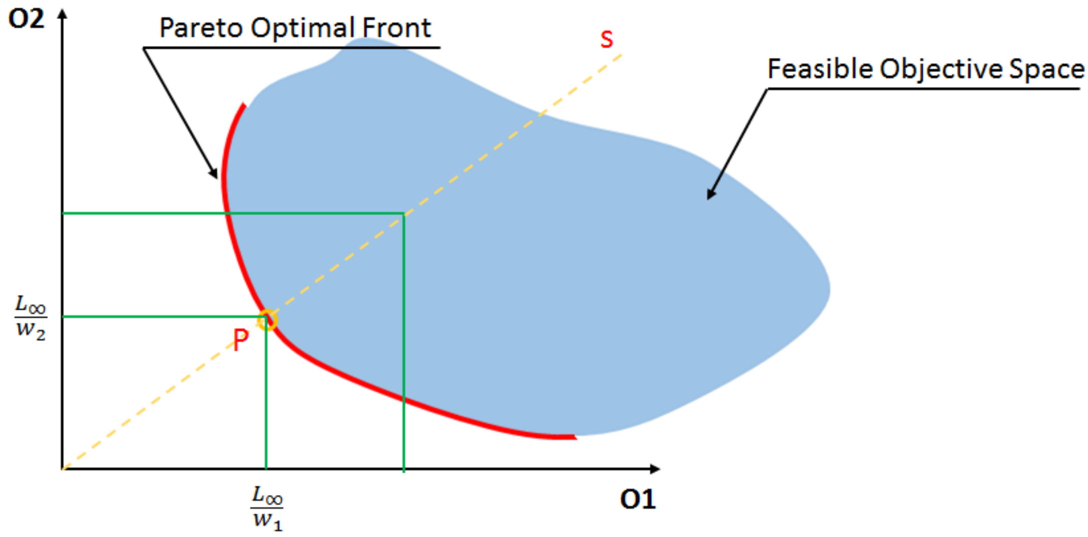


Figure 6-8 the Weighted Metric Method with $p = \infty$

6.5.3.2 Independent Sampling

Figure 6-9 illustrates the method of getting a Pareto-optimal front by weighted min-max approach. Here $\vec{w}_m$ are independent weight vectors of two objectives. Different from Norm1 (weighted sum method) and Norm2 which need to find the tangent points of contour lines or ovals with the research space, the weighted min-max approach could easily be described as searching for the interception points of $\vec{w}_m$ and feasible objective space. The figure could be more intuitive and understandable.

As a Pareto front could be regards as a set of continuous points. To get the Pareto-optimal front, we use different weights for each objective and each weight could get an optimal point with the help of OptQuest Engine which is embedded in AnyLogic. The more sampling be used, the closer we are to the real Pareto-optimal solution front. But because it is impossible and also unnecessary to optimize the endless solutions, only about dozens of sets of weight will be optimized in this thesis to get the trends of the Pareto-optimal solution front.

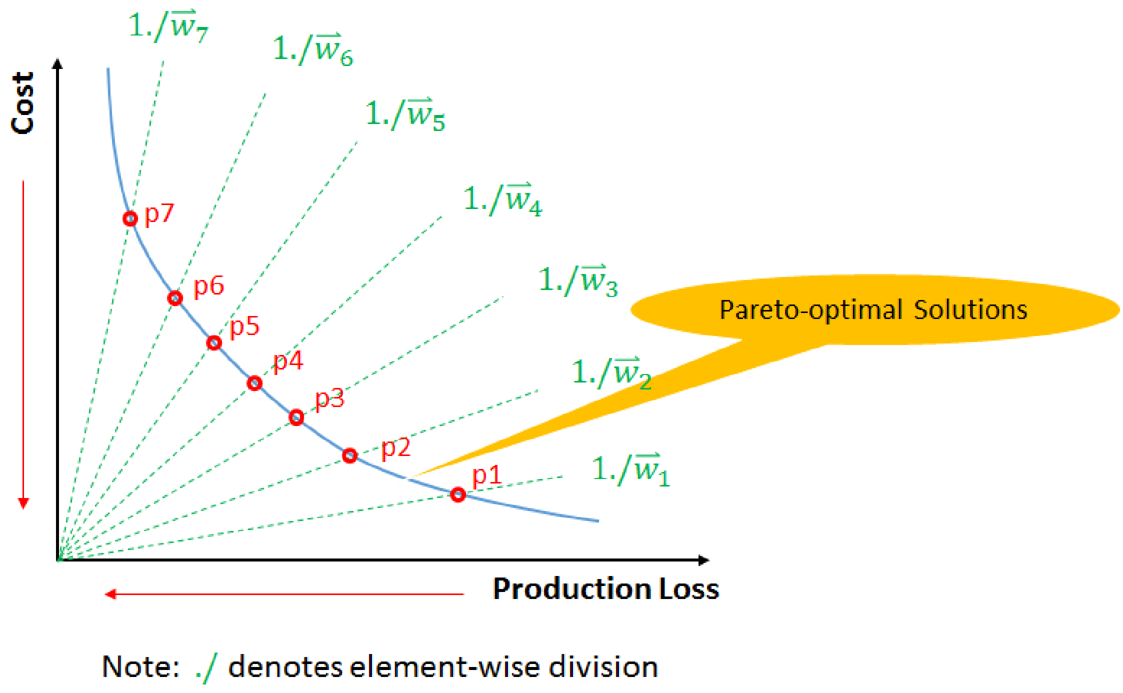


Figure 6-9 Weighted Min-Max Approach

Back to the assembly case in this thesis, see Figure 6-9, p1 to p7 are seven points on the ideal Pareto-optimal front and $\vec{w}_1$ to $\vec{w}_7$ stand for the different weight vectors of two objectives. It could be easily found that the weight of production loss for p7 is much more than that for p1. In other words, in solution p7, the decision maker cares much more about the production rate and is willing to pay more for a low production loss. By the same token, for solution p1, saving money is more important than the number of aircrafts. But no matter which point on the Pareto optimal front, it is the optimal solution based on that weight of two objectives.

6.5.3.3 Objective Function

As subsection 6.5.3.1 mentioned, the main idea of weighted approach is to combine the objectives into one and optimize that objective. Here equation (6-13) are the two objectives in this aircraft assembly case where $O1$ and $O2$ stand for production loss and maintenance cost respectively. Vector $\vec{x}$ represents the maintenance threshold ($V_i^{mtn},\ i = 0,1,2$) and response time ($T_i^{Resp},\ i = 0,1,2$) of each machine as equation (6-14) shown.

$$f_1(x) = R^{prodLoss} = O1$$
$$f_2(x) = C^{mtn} = O2 \tag{6-13}$$

$$\vec{x} = \begin{bmatrix} V_0^{mtn} \\ T_0^{Resp} \\ V_1^{mtn} \\ T_1^{Resp} \\ V_2^{mtn} \\ T_2^{Resp} \end{bmatrix} \tag{6-14}$$

According to equation (6-12), which is the general function of the weighted min-max approach, the objective function in this case could easily be given as (6-15) shown.

$$f(x) = max\{w_1 O1, w_2 O2\} \tag{6-15}$$

7 RESULTS

7.1 Performance Trade-Off

All the simulation and optimization results are calculated by the commercial simulation software AnyLogic and the optimization engine OptQuest which is embedded in AnyLogic. The optimization is based on the scatter search methodology which has been introduced in section 6.4. The optimization began with 40 interactions and 20 replications pre interaction. After getting a rough result, the optimization parameter change to 100 replications pre interaction and get the final results.

7.1.1 Simulation and Optimization with 20 Replications

Optimization with 40 interactions means the optimization engine runs 40 rounds to search the optimal result. Looking back to section 6.4, here 40 is the value of Max_{Iter} in Figure 6-4. Though with more rounds of running would have more opportunities to get a better solution, it is not so necessary. As Figure 7-1 shown, after about 15 iterations (X axis), there is no significant change in the optimization results (Y axis). Therefore $Max_{Iter} = 40$ is sufficiently high in this case.

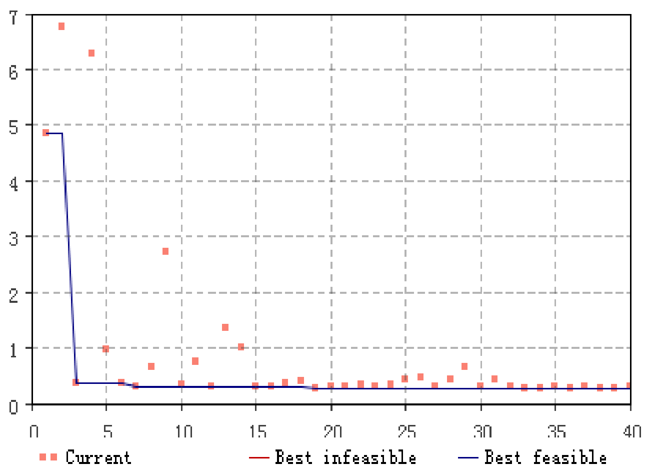


Figure 7-1 Optimization Interactions

For each interaction, 20 replications means for each group of input, the simulation model will run 20 times to get mean values as the output as Figure 6-3 shown (Page 47). The reason of multiple sampling is the existence of uncertainties in the system (section 6.3) and the stochastic data will be further discussed later in section 7.2. After comparing the results of different replications, it is found that 20 replications are not enough to get an ideal result. But even for an optimization with 40 interactions and 20 replications pre interaction, the simulation model need to run about eight hundreds runs to get the optimal results. So as a kind of initial experiment, we use only 20 replications to see the trend of the results.

Table 7-1 Results with 20 Replications (Weight from 0.1 to 0.9)

Weight		$Machine$	V_i^{mtn}	T_i^{Resp}	$R^{prodLoss}$	C^{mtn}	$N^{Aircraft}$	N_i^{mtn}
$O1$	$O2$							
0.1	0.9	Crane	415.56	285.78	0.478	1.27	814.33	55.36
		FlexTrack	129.24	412.5				18.524
		RivetGun	737.82	476.64				21.604
0.2	0.8	Crane	414.6	265.5	0.454	1.284	815.33	56.19
		FlexTrack	104.28	447.9				17.97
		RivetGun	703.26	352.44				21.06
0.3	0.7	Crane	429.24	270.96	0.476	1.306	814.41	57.82
		FlexTrack	111.48	450.36				18.01
		RivetGun	737.1	347.04				22.06
0.4	0.6	Crane	414.42	269.16	0.469	1.306	814.72	57.75
		FlexTrack	100.5	409.92				18.03
		RivetGun	735.72	353.1				22.14
0.5	0.5	Crane	345.9	201.9	0.469	1.3	814.72	49.51
		FlexTrack	115.44	366.78				18.34
		RivetGun	632.52	471.84				18.84
0.6	0.4	Crane	413.28	263.46	0.45	1.248	815.53	55.54
		FlexTrack	66.96	468.12				17.04
		RivetGun	747.78	369.36				22.26
0.7	0.3	Crane	414.6	265.5	0.462	1.278	815.01	55.95
		FlexTrack	106.98	447.9				17.91
		RivetGun	703.26	352.44				20.91
0.8	0.2	Crane	386.52	139.68	0.406	1.463	817.33	56.59
		FlexTrack	35.58	452.04				16.5
		RivetGun	739.26	475.26				21.93
0.9	0.1	Crane	391.08	139.92	0.378	1.614	818.52	57.47
		FlexTrack	262.8	451.38				22.48
		RivetGun	450.96	351.54				15.69

Table 7-1 illustrates the simulation and optimization results with 20 replications under the weights from $\vec{w}_1 = (0.1, 0.9)$ to $\vec{w}_9 = (0.9, 0.1)$ and Figure 7-2 shows the Pareto optimal front. Here $R^{prodLoss}$ means the production loss and C^{mtn} is the maintenance cost. Different from the ideal Pareto optimal front as Figure 6-9 shown, except $\vec{w}_8 = (0.8, 0.2)$ and $\vec{w}_9 = (0.9, 0.1)$, the values of maintenance cost with different weights are similar and irregular to each other. Obviously, this result is not the one we supposed to have.

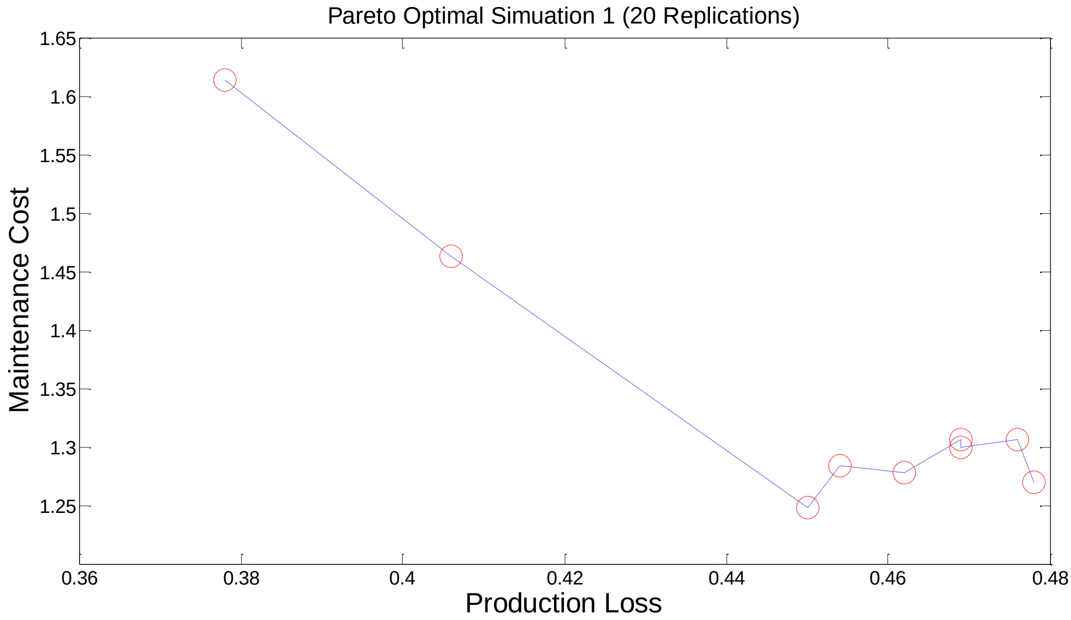


Figure 7-2 Pareto Optimal Solutions with 20 Replications (0.1 ~ 0.9)

To find out the reason for these unsatisfied result, we should look back to equation (6-15), which is the optimization objective for the OptQuest engine. For each $\vec{w}_i = (w_1, w_2)$, the optimization objective is $max\{w_1 O1, w_2 O2\}$.But look at the results in Table 7-1, the values of $O2$ (C^{mtn}) in the nine groups are all much bigger than $O1$ ($R^{prodLoss}$). That means the value of $w_2 O2$ trends to be bigger than the value of $w_1 O1$ unless w_1 is bigger than w_2 (like $\vec{w}_8$ and $\vec{w}_9$). So in most of the time, $O1$ does not really work in the optimization processes. Changing the normalizing constant C_c^{mtn} (equation (6-5) in section 6.2.2) is one of the methods to deal with this problem. But this method need to rerun all the optimization again and still not sure whether the new C_c^{mtn} will have the same problem again or not. Another method is keeping these data and running more groups of optimization with different weight. Because in theory, the Pareto optimal solutions could be found by optimizing endless number of different weights, but some weights maybe not really work as $\vec{w}_{e1}$ and $\vec{w}_{e2}$ shown in Figure 7-3. Some weights like $\vec{w}_{e4}$, $\vec{w}_{e5}$ and $\vec{w}_{e6}$ work well but the gaps between them are too wide. To get the Pareto optimal front, more weights need to be optimized to make up these gaps. It is easy to find from Figure 7-2 and Table 7-1 that there are two big gaps between $\vec{w}_7 = (0.7, 0.3)$, $\vec{w}_8 = (0.8, 0.2)$ and $\vec{w}_9 = (0.9, 0.1)$.

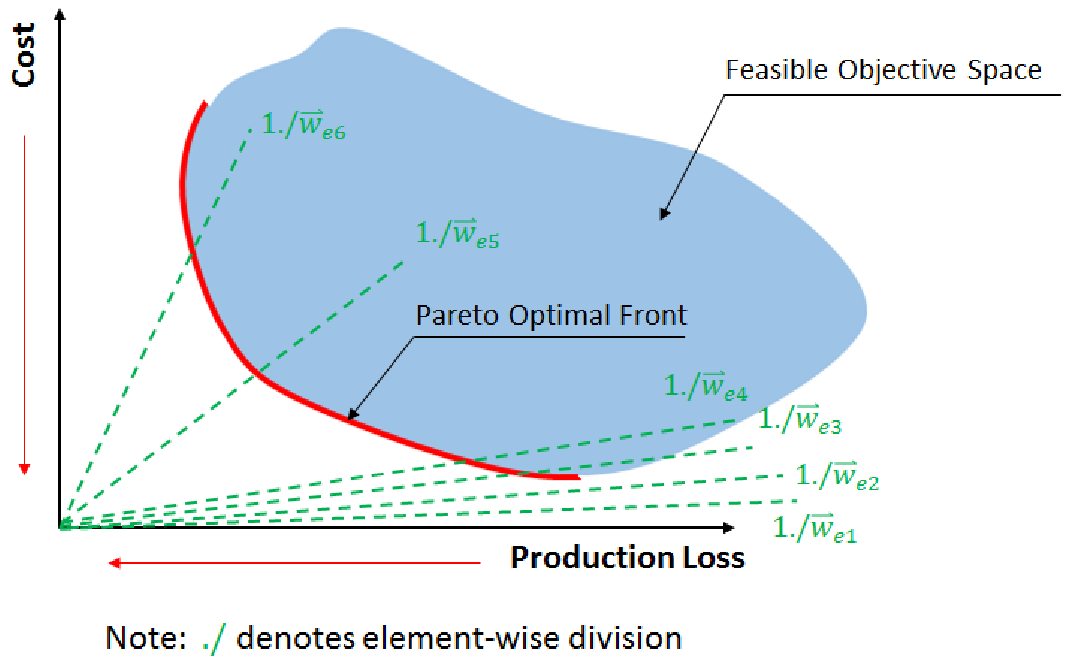


Figure 7-3 Example of Unbalanced Weights Distribution.

Table 7-2 and Figure 7-4 demonstrates the results of the simulation and optimization with 20 replications after running more groups of weights. Figure 7-5 zooms in a part of Figure 7-4 and depicts the results from weight $(0.5, 0.5)$ to $(0.975, 0.025)$. From this figure we could roughly find that as the decrease of production loss, the maintenance cost keep on increasing. After some points like P_{985} and P_{99} in Figure 7-4, where the weights are $(0.985, 0.015)$ and $(0.99, 0.01)$ respectively, even a very small decrease of production loss will cost a large increase of maintenance cost. It could give the decision makers such kind of sign that unless really necessary, usually it is not worthy to keep the production loss in such a low value.

Though the simulation and optimization results with 20 replications could give a trend of the Pareto optimal front, the trend is not clear enough. Because the sampling number is not big enough to get a stable mean outputs (see Figure 6-3 in section 6.3.2). This part would be further discussed later in section 7.2. The aim of running the 20 replications is to find the suitable weights as the data preparation for the final results.

Table 7-2 Results with 20 Replications (Weight from 0.5 to 0.99)

Weight $O1$	Weight $O2$	$Machine$	V_i^{mtn}	T_i^{Resp}	$R^{prodLoss}$	C^{mtn}	$N^{Aircraft}$	N_i^{mtn}
0.5	0.5	Crane	345.9	201.9	0.469	1.3	814.72	49.51
		FlexTrack	115.44	366.78				18.34
		RivetGun	632.52	471.84				18.84
0.6	0.4	Crane	413.28	263.46	0.45	1.248	815.53	55.54
		FlexTrack	66.96	468.12				17.04
		RivetGun	747.78	369.36				22.26
0.7	0.3	Crane	414.6	265.5	0.462	1.278	815.01	55.95
		FlexTrack	106.98	447.9				17.91
		RivetGun	703.26	352.44				20.91
0.75	0.25	Crane	392.88	246.48	0.416	1.356	816.93	53.33
		FlexTrack	49.56	275.34				17.09
		RivetGun	736.98	475.2				21.67
0.8	0.2	Crane	386.52	139.68	0.406	1.463	817.33	56.59
		FlexTrack	35.58	452.04				16.5
		RivetGun	739.26	475.26				21.93
0.85	0.15	Crane	268.92	419.64	0.388	1.701	818.08	37.69
		FlexTrack	559.14	384.12				41.43
		RivetGun	237.12	180.96				12.87
0.9	0.1	Crane	391.08	139.92	0.378	1.614	818.52	57.47
		FlexTrack	262.8	451.38				22.48
		RivetGun	450.96	351.54				15.69
0.95	0.05	Crane	391.92	139.92	0.384	1.711	818.23	57.23
		FlexTrack	330.84	451.38				25.06
		RivetGun	468.48	243.06				16.23
0.955	0.045	Crane	375.42	149.4	0.381	1.731	818.38	54.51
		FlexTrack	399.18	423.3				28.44
		RivetGun	325.44	241.92				13.96
0.965	0.035	Crane	337.62	304.92	0.42	1.6	816.76	45.75
		FlexTrack	433.5	326.34				31.06
		RivetGun	263.1	220.62				13.32
0.975	0.025	Crane	123.12	134.94	0.349	2.645	819.73	33.27
		FlexTrack	707.46	372.72				69.19
		RivetGun	526.68	446.88				16.89
0.985	0.015	Crane	51.3	173.64	0.333	11.098	820.38	29.63
		FlexTrack	183.18	11.28				20.95
		RivetGun	75.3	70.2				11.57
0.99	0.01	Crane	102.9	275.04	0.258	14.255	823.54	30.66
		FlexTrack	338.7	12.12				27.06
		RivetGun	160.74	13.32				12.24

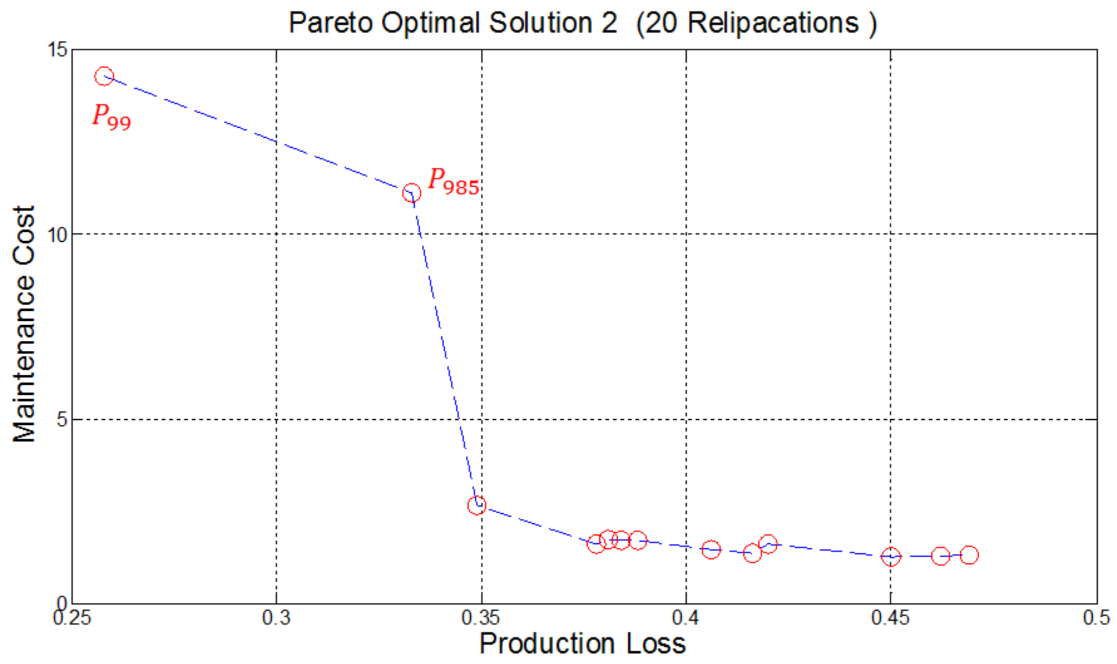


Figure 7-4 Pareto Optimal Solutions with 20 Replications (0.5 ~ 0.99)

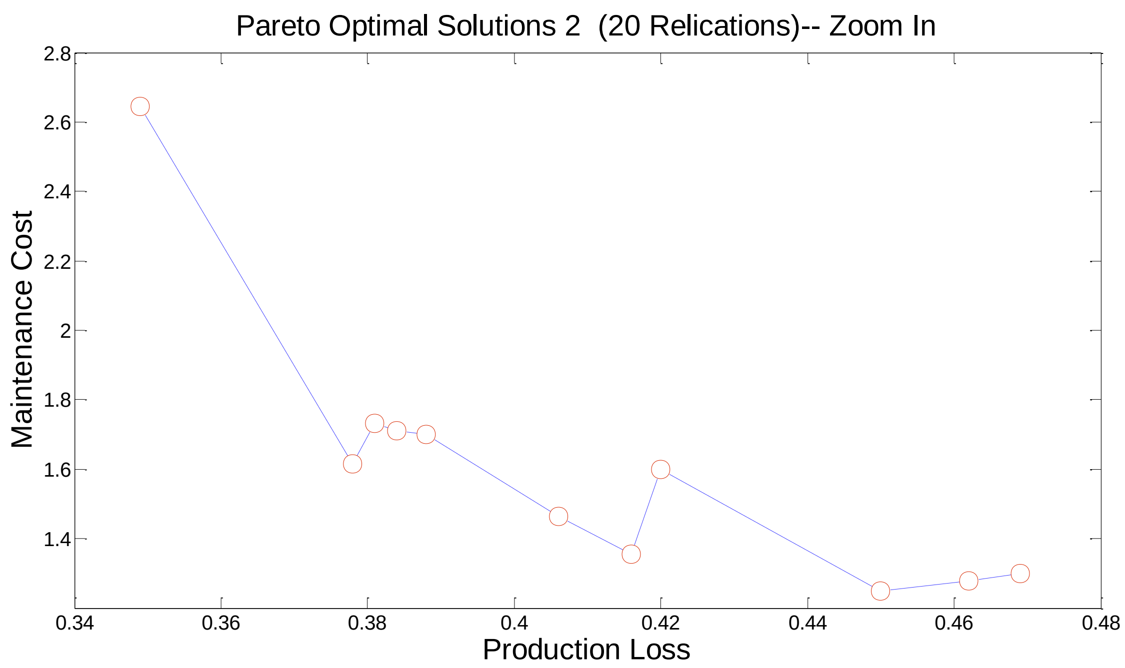


Figure 7-5 Pareto Optimal Solutions with 20 Replications (0.5 ~ 0.975)

7.1.2 Simulation and Optimization with 100 Replications

Figure 7-6 is the Pareto optimal solution of the simulation and optimization with 100 replications. The detailed results are shown in Table 7-3, whose weights are roughly based on the data of 20 replications. The last weight $(0.99, 0.01)$ here is a kind of limitation of these weights. Usually this weight will not be used in real production processes and the reason is quite clear from the data. While the production loss is already at a very high weight (like weight $(0.975, 0.025)$ here), even a small decrease of the production loss (from 0.339 to 0.33) will cost a much sharper increase to the maintenance cost (from 2.171 to 7.572).It is an important information for the decision makers to avoid unnecessary investment on the extra maintenance cost.

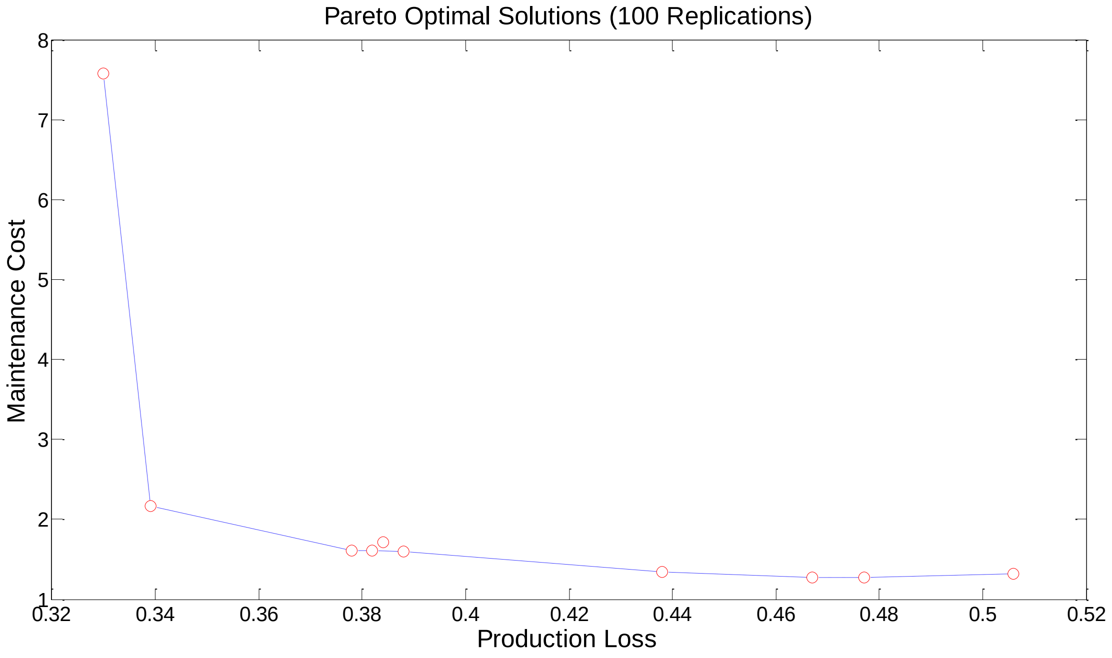


Figure 7-6 Pareto Optimal Solutions with 100 Replications

Figure 7-7 depicts the solutions from weight $(0.5, 0.5)$ to $(0.975, 0.025)$ which is zoomed in from Figure 7-6. We can find that the Pareto optimal front in Figure

7-7 is smoother than that of Figure 7-5 and the trend of it is much clearer as well. This demonstrates that by increasing the replication number, the quality of the data could be improved obviously.

Look at the figure, the shape of this Pareto Optimal front is quite close to the idea one but the solution $P_{0.95}$, whose weight is $(0.95\,, 0.05)$, is not very ideal as it is a little defected from the other solutions. The $P_{0.95}$ could be regarded as a solution in the feasible objective space and close to the Pareto Optimal front but not really cover the front. Optimization results with the same weight and parameters usually do not have exactly the same value. Most of the time, these solutions are very close to each other but still have tiny difference because the existence of the randomness. But this will not affect the overall trend and conclusions of this work.

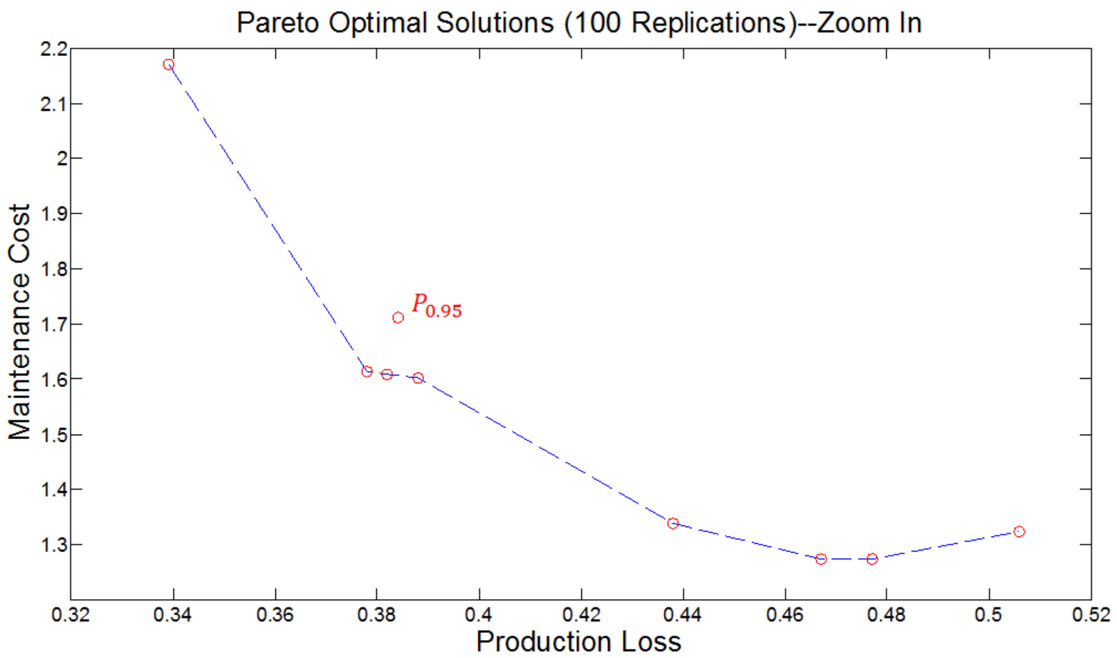


Figure 7-7 Pareto Optimal Solutions with 100 Replications (Zoom In)

From the results, we could also find that the maintenance cost ($O2$) approximately cannot be lower than 1.2 even we deliberately set the weight of $O2$ bigger than 0.5 (see Table 7-1). It could be regarded as the optimal boundary for the maintenance cost. The reason is that no matter what the values of maintenance threshold and response time are, no matter the maintenance is performed before or after the breakdown, the degradation of assembly machines is always essential and the maintenance is always necessary. In addition, the maintenance cost included not only the response cost, but also the fixed cost. These required minimum maintenance services will at least incur the fixed maintenance cost. These explain why the maintenance cost cannot be lower than a certain level, and effectively determine the boundary of the objective space.

Table 7-3 Simulation and Optimization Results with 100 Replications

Weight		$Machine$	V_i^{mtn}	T_i^{Resp}	$R^{prodLoss}$	C^{mtn}	$N^{Aircraft}$	N_i^{mtn}
$O1$	$O2$							
0.5	0.5	Crane	415.56	285.78	**0.477**	**1.274**	814.38	55.49
		FlexTrack	412.5	129.24				18.58
		RivetGun	737.82	476.64				21.66
0.6	0.4	Crane	419.4	273.24	**0.467**	**1.273**	814.8	56.51
		FlexTrack	450.66	133.14				18.6
		RivetGun	720.42	476.7				21.05
0.7	0.3	Crane	407.16	213.9	**0.506**	**1.323**	813.16	56.7
		FlexTrack	456.9	97.8				17.72
		RivetGun	750.18	477.24				21.85
0.75	0.25	Crane	405.06	265.5	**0.438**	**1.338**	816.01	54.81
		FlexTrack	310.2	76.8				17.71
		RivetGun	737.76	475.5				21.72
0.8	0.2	Crane	392.34	139.32	**0.388**	**1.602**	818.1	57.79
		FlexTrack	451.86	282				22.96
		RivetGun	500.46	265.8				16.62
0.85	0.15	Crane	390.54	139.8	**0.382**	**1.608**	818.36	57.36
		FlexTrack	452.7	254.76				22.19
		RivetGun	458.46	337.02				15.81
0.9	0.1	Crane	391.08	139.92	**0.378**	**1.614**	818.52	57.47
		FlexTrack	451.38	262.8				22.48
		RivetGun	450.96	351.54				15.69
0.95	0.05	Crane	391.92	139.92	**0.384**	**1.711**	818.23	57.23
		FlexTrack	451.38	330.84				25.06
		RivetGun	468.48	243.06				16.23
0.975	0.025	Crane	192.6	134.94	**0.339**	**2.171**	820.13	37.55
		FlexTrack	622.68	372.72				50.17
		RivetGun	612.42	448.26				18.5
0.99	0.01	Crane	67.74	90	**0.33**	**7.572**	820.53	31.27
		FlexTrack	130.8	19.68				19.55
		RivetGun	101.04	14.1				11.74

In a word, as this thesis discussed before, we could not get the minimal production loss and maintenance cost at the same time, but with the help of Pareto optimal front, we could find an optimal solution at a certain weight of these two objectives. The simulation and optimization results illustrate the optimal relationship between the production loss and maintenance cost. The decrease of production loss will increase the maintenance cost and this Pareto optimal front gives decision makes a clear map to use the suitable strategies under different situations of the company.

7.2 Monte-Carlo Simulation

Monte Carlo experiment is a type of computational algorithm that rely on repeated random sampling to obtain and display a collection of simulation outputs for a stochastic model or for a model with stochastically varied parameter(s) [60]. In this thesis, a Monte Carlo experiment is used to evaluate whether the obtained optimal solution is the performance expected from the system and how sensitive the design solution is to the uncertainties.

In this ARJ21 assembly model, the uncertainties come from three areas, which are the assembly working processes, the automatic machines and the processes of waiting or doing the maintenance. Because of the existence of randomness, each run would produce a different output (product loss and maintenance cost) even if the input parameters (maintenance threshold and response time) keep the same. In this case, while running the simulation and optimization, each group of inputs runs 100 replications and gets 100 groups of different outputs. The optimization engine will use the mean value of the outputs as a basis for the next run of optimization until the Max_{Iter} reaches. The optimization experiment in AnyLogic can provide the definite value of the inputs (i.e. V_i^{mtn} and T_i^{Resp} $, i = 0,1,2$) while the optimization objective (i.e. $max\{w_1 O1, w_2 O2\}$) reaches the minimal value at a certain weight vector. This minimal value, which equals to the optimization function $f(x) = max\{w_1 O1, w_2 O2\}$ (equation (6-15)), could be shown at the same time but do not represent the value of $O1$ and $O2$ separately. So in this case, the production loss and maintenance cost in Table 7-1,

Table 7-2, and Table 7-3 are all calculated by Monte Carlo experiment.

To demonstrate the meaning of the numbers in the results in detail, here we chose one group of the data ($\vec{w} = (0.85\,, 0.15)$) as example. Figure 7-8, Figure 7-9 and Figure 7-10 depict the distribution of production loss ($R^{prodLoss}$), number of aircraft ($N^{Aircraft}$) and maintenance cost (C^{mtn}) respectively while the weight of the two optimal objectives is $\vec{w} = (0.85\,, 0.15)$. The 100 runs of simulation share the same group of inputs which is shown in Table 7-3. The figures are drawn by MATLAB with the data from Monte Carlo experiment in AnyLogic, which can show the detailed results of the 100 replications.

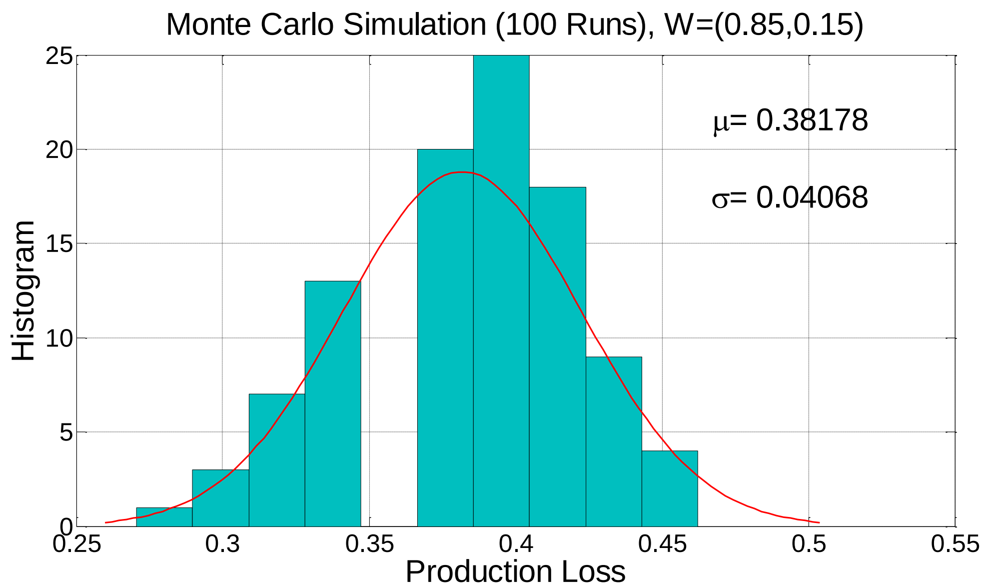


Figure 7-8 Monte Carlo Simulation Results of Production Loss

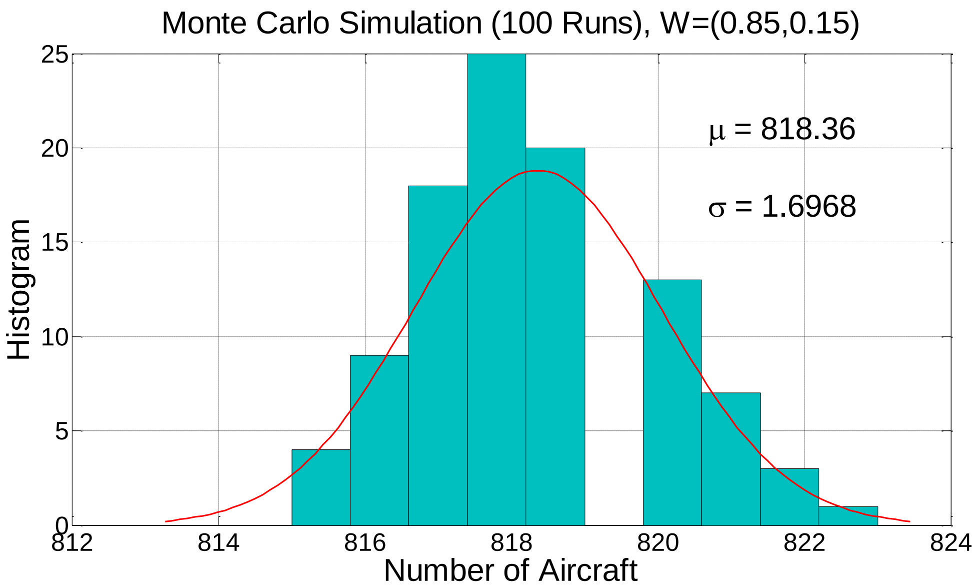


Figure 7-9 Monte Carlo Simulation Results of Number of Aircraft

From the data of Figure 7-8 and Figure 7-9, we can find that the mean value (μ) of production loss is about 0.382 and that of the number of aircraft is about 818.36 aircrafts in ten years. Comparing with the maximum number of the aircraft that COMAC could manufacture, which is about 834.29 aircrafts (see equation (6-1) in section 6.2.1) without considering the loss from maintenance, the loss is about 15.93 aircrafts in ten years. The production loss value ($R^{prodLoss} = 0.382$) is the rate of the real loss number and the acceptable loss number ($R^{AcPL} \times N^{max} = 5\% \times 834.29$) which is defined by equation (6-2) in section 6.2.1.

The distributions follow the normal deviation and the standard deviation values (σ) of production loss and number of aircraft are 0.04068 and 1.6968, respectively. That means about 68.2% of the cases the production loss will be between ± 1.7 aircrafts in ten years, which equals ± 0.17 aircrafts per year from the expected value.

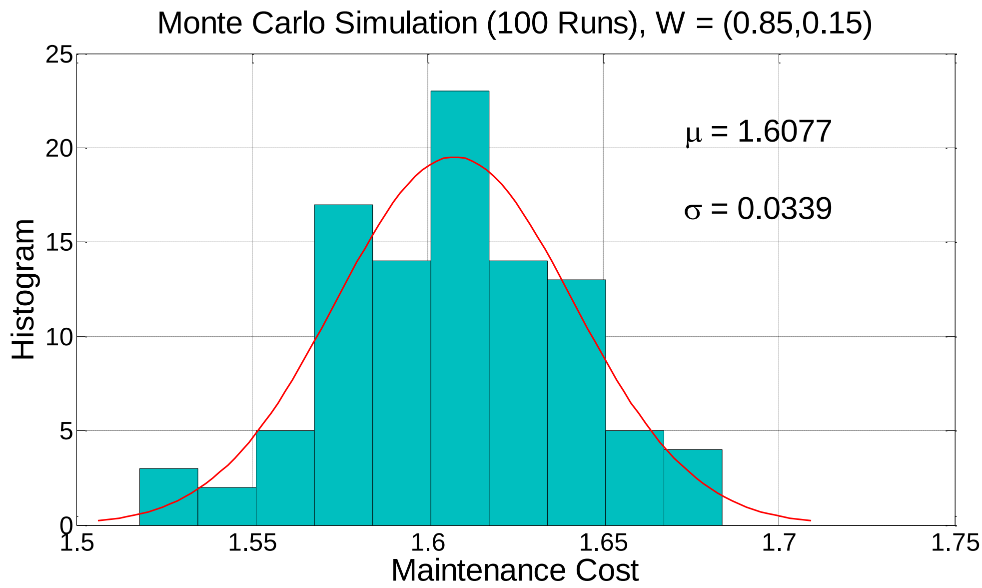
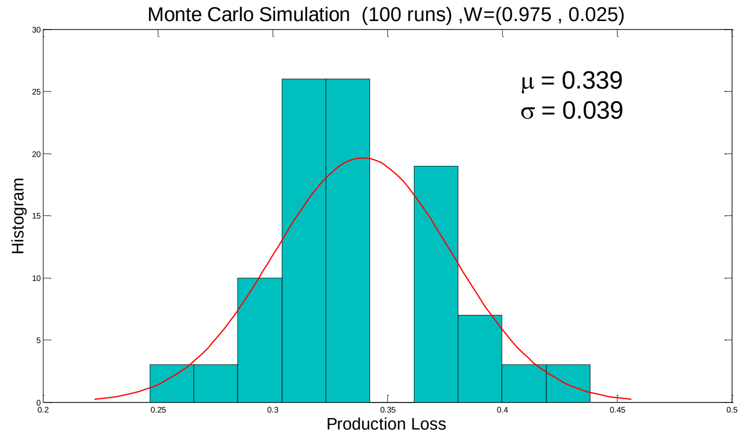


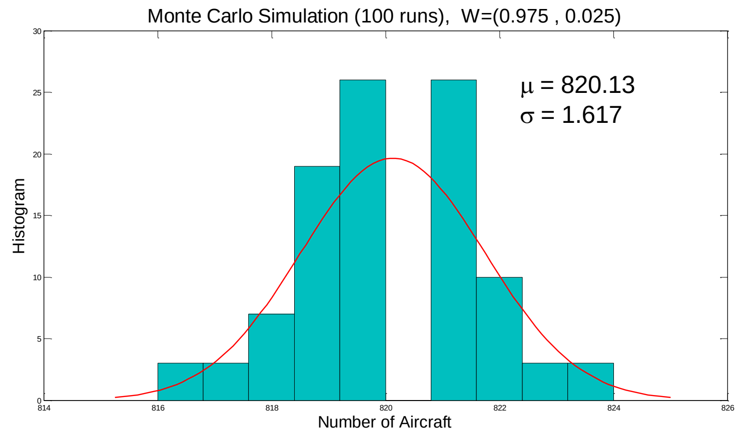
Figure 7-10 Monte Carlo Simulation Results of Maintenance Cost

Maintenance cost (C^{mtn}) in this thesis could be regarded as a rate to the standard maintenance cost (or normalizing constant). While the weight is $\vec{w} = (0.85\,, 0.15)$, the mean value of maintenance cost is 1.61 and the standard deviation is about 0.03 as Figure 7-10 shown.

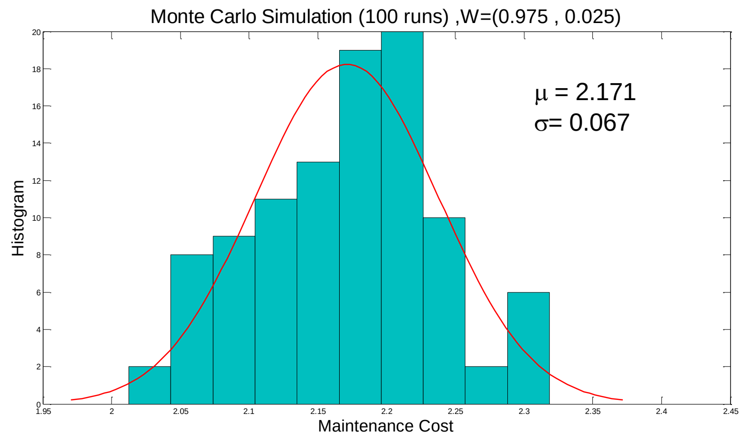
To deliberately demonstrate that $\vec{w} = (0.85\,, 0.15)$ is not a special sample, here we post the Monte Carlo simulation solutions with $\vec{w} = (0.975\,, 0.025)$ as well (see Figure 7-11). From these results, we could find that Monte Carlo simulation results follow the normal deviation and the uncertainty of $\pm 25\%$ do not significantly affect the results, which means low sensitivities for the designed model.



(a) Production Loss



(b) Number of Aircraft



(c) Maintenance Cost

Figure 7-11 Monte Carlo Simulation Results with $\vec{w} = (\mathbf{0.975}\,,\ \mathbf{0.025})$

8 CONCLUSIONS AND FUTURE WORK

In a current competitive environment, increasing production rate and reducing costs are the key drivers in aircraft manufacturing. More (semi-)automatic assembly machines have increasingly being used in the aircraft assembly lines as a mean to deliver high production rate while meeting high quality requirements. However, the production throughput is effectively depend on the operational availability of these machines. Integration of CBM into the assembly system has potential benefits as a way to minimize the production loss and maintenance related cost. Maintenance are performed as needed, hence avoiding unnecessary downtime and maintenance cost.

In a CBM enabled aircraft assembly system, there are self-active interactions between the subsystems, e.g. CM system self-triggers a maintenance order when the system reliability falls the below a certain level. This example of active self-aware behaviour cannot straightforwardly be modelled using DESs. In this case, where, besides the assembly process, independent entities are in addition parts of the system, ABS is proved effective as it allows complex active interactions between entities to be naturally captured.

Production rate and maintenance cost are the competing objectives in an integrated CBM aircraft assembly system. Finding trade-offs between the production rate and maintenance cost is equivalent to finding a Pareto optimal surface. The conventional non-dominated ranking methods will not be practically feasible due to the computational burden required by the Monte-Carlo simulation. This limitation can be addressed by independently sampling the Pareto surface using the Weighted Min-Max method. The approach allows less number of populations to be used in the optimization as it does not need to probe the whole Pareto front, and hence effectively a reduction in the computation intensity required.

In our ARJ21 case study, the preventive maintenance threshold and required service level are the key design parameters that determine the overall performance of the assembly system. Because of uncertainties, increase production rate will require a high required service level (i.e. fast response time) to avoid breakdowns before the maintenance is performed, and consequently this will increase the maintenance cost and sometimes can be significant. However, compromising on the production rate does not always mean a further decrease in maintenance cost. The minimum cost is from the actually cost in maintaining the machines and the minimum service level. Pareto surface is an important piece of information to the system designer. Together with Monte-Carlo simulation, it can be used to support decision making in terms of cost-benefit of different design solutions and also what could be achieved.

In this example, even though in small scale, it can be seen that CBM has potential to be applicable in (semi-)automatic aircraft assembly lines. However, there are still many work could be continued in the future. In this thesis, we only consider one assembly process which is the final joint assembly. Multiple assembly processes could be studied in the future. For example, while two (or more) aircrafts are being assembled at the same time, they may need to share some of the equipment. How to plan the utilization of equipment could be studied further more via simulation and optimization. Furthermore, until now the developed ABS assembly simulation has not been validated against the real assembly data due to the fact that, when this study was conducted, there is no actual assembly data (which is commercially in confidence) released by COMAC. However, the validation of this model needs to be further carried out to ensure the correctness of the model when the field data become available. Moreover, in terms of optimization, other different optimization methods like genetic algorithms (GAs), simulated annealing and teaching-learning-based optimization (TLBO) should also be used in the optimization to by comparison ensure the true Pareto font is found and consequently their performance in terms of computation and solution can be compared and analysed. On the other hand, this thesis only focuses on the Condition Based maintenance and demonstrates it could be applied in aircraft assembly process. But there still have other maintenance strategies like Predictive Maintenance. Further study could try some other regimes and make comparisons between them. Triangular distribution is applied in this thesis to present the uncertainty. Whether different distributions will affect or not affect the trade-off between objectives could be experimented by simulation and optimization in the future.

REFERENCES

[1] MTorres, “Total Solution for Aircraft Automatic Assembly Jigs,” in *MTorres*, California, 2013.

[2] P. Lute, “An Investigation of Airbus A380 Stage 01 Wing Box Assembly Using Discrete Event Simulation,” Cranfield University, Bedfordshire, 2007.

[3] Electroimpact Inc., “A380 Stage 1 GRAWDE Machine,” Electroimpact Inc., Mukilteo,WA, 2003.

[4] IBM Corporation, “Predictive Maintenance for Manufacturing,” 2011.

[5] Wikipedia, “Preventive Maintenance,” Wikipedia, [Online]. Available: http://en.wikipedia.org/wiki/Preventive_maintenance. [Accessed 15 May 2013].

[6] ResolveFM, “Preventive / Corrective Maintenance,” ResolveFM, [Online]. Available: www.resolve.com.au. [Accessed 22 Apr 2013].

[7] A. Kelly, Maintenance Strategy, Oxford: Butterworth-Heinemann, 1997.

[8] NACE, “Maintenance Strategies,” www.nace.org, 15 May 2013. [Online]. Available: http://events.nace.org/library/corrosion/Inspection/Strategies.asp. [Accessed 15 May 2013].

[9] I. Grigoryev, AnyLogic 6 in Three Days: A Quick Course in Simulation Modeling, Anylogic North America, 2012.

[10] J.-B. Leger, E. Neunreuthe, B. Iung and G. Morel, “Integration of the Predictive Maintenance in Manufacturing System,” in *Advanced in Manufacturing*, Vandoeuvre, Springer London, 1999, pp. 133-144.

[11] Z. Tian, D. Lin and B. Wu, “Condition Based Maintenance Optimization Considering Multiple Objectives,” *Journal of Intelligent Manufacturing,* vol. 23, no. 2, pp. 333-340, 2009.

[12] Y. Jin, Z. Jiang and W. Hou, “Multi-objective integrated optimization research on preventive maintenance planning and production scheduling for a single machine,” *The International Journal of Advanced Manufacturing Technology,* vol. 39, no. 9-10, pp. 954-964, 2008.

[13] D. Achermann, “Modelling,Simulation and Optimization of Maintenance Strategies under Consideration of Logistic Processes,” Menzna LU, 2008.

[14] Š. Valčuha, A. Goti, J. Úradníček and I. Navarro, “Multi-equipment condition based maintenance optimization by multi- objective genetic algorithm,” *Journal of Achievements in Materials and Manufacturing Engineering,* vol. 45, no. 2, pp. 188-193, 2011.

[15] L. Tautou and H. Pierreval, “Using Evolutionary Algorithms and Simulation for the Optimization of Manufacturing Systems,” *IIE Transactions,* vol. 29, no. 3, pp. 181-189, 1997.

[16] D. Baglee, “Maintenance Strategy Development in the UK Food and Drink Industry,” *International Journal of Strategic Engineering Asset Management,* 2013.

[17] J. Reimann, G. Kacprzynski, D. Cabral and R. Marini, “Using Condition Based Maintenance to Improve the Profitability of Performance Based Logistic Contracts,” in *Annual Conference of the Prognostics and Health Management Society*, 2009.

[18] A. Grall , C. Bérenguer and L. Dieulle, “A Condition-based Maintenance Policy for Stochastically Deteriorating Systems,” *Reliability Engineering and System Safety,* vol. 76, no. 2, pp. 167-180, 2002.

[19] M. Rolón and E. Martínez, "Agent-based modeling and simulation of an autonomic manufacturing execution system," *Computers in Industry,* vol. 63, no. 1, pp. 53-78, 2012.

[20] W. Shen, Q. Hao, H. J. Yoon and D. H. Norrie , "Applications of agent-based systems in intelligent manufacturing: An updated review: An updated review," *Advanced Engineering Informatics,* vol. 20, no. 4, pp. 415-431, 2006.

[21] L. Holst, Integrating Discrete-Event Simulation into the Manufacturing System Development Process, Lund, Sweden: Division of Robotics, 2001.

[22] J. A. B. Montevechi, R. d. C. Miranda and J. D. Friend, "Sensitivity Analysis in Discrete-Event Simulation Using Design of Experiments," in *Discrete Event Simulations-Development and Applications*, Croatia, InTech, 2012.

[23] Y. Carson and A. Maria, "Simulation Optimization: Methods and Applications," in *Simulation Conference*, Atlanta, Georgia, USA, 1997.

[24] R. Marti and M. Leguna, "Scatter Search: Basic Design and Advanced Strategies," *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial,* pp. 123-130, 2003.

[25] A. Ghosh and S. Dehuri, "Evolution Algorithms for Multi-Criterion Optimization: A Survey," *International Journey of Computing & Information Sciences,* vol. 2, no. 1, 2004.

[26] M. Laguna, R. Martí, M. Gallego and A. Duarte, "The Scatter Search Methodology," in *Wiley Encyclopedia of Operations Research and Management Science*, Hoboken, New Jersey, Wiley-Blackwell , 2011.

[27] COMAC, "ARJ21 Regional Jet Program," 7 September 2013. [Online]. Available: http://english.comac.cc/products/rj/pi2/index.shtml.

[28] O. Rüscher and H. Mayländer, “Automated Alignment and Marry-Up of Aircraft Fuselage Sections with a Final Assembly Line,” in *SAE Aerospace Automated Fastening Conference*, Washington, 2001.

[29] J. Shang, T. Sattar, S. Chen and B. Bridge, “Design of a climbing robot for inspecting aircraft wings and fuselage,” *Industrial Robot: An International Journal,* vol. 34, no. 6, pp. 495-502, 2007.

[30] Electroimpact, “Flex Track,” Electroimpact, Mukilteo.WA, 2013.

[31] IEEE/PES Task Force on Impact of Maintenance Strategy on Reliability of the Reliability, Risk and Probability Applications Subcommittee, “The Present Status of Maintenance Strateties and the Impact of Maintenance on Reliability,” *IEEE Transactions on Power Systems,* vol. 16, no. 4, pp. 638-646, 2001.

[32] P. D. Van, . A. Voisin, E. Levrat and B. Iung, “Condition-Based Maintenance with both Perfect and Imperfect Maintenance Actions,” in *Annual Conference of the Prognostics and Health Management Society 2012*, Minneapolis, Minnesota, USA, 2012.

[33] H. Chouikhi, A. Khatab and R. Nidhal , “Condition-Based Maintenance for Availablity Optimization of Productions System Under Environment Constraints,” in *9th International Conference on Modeling, Optimization & SIMulation*, Bordeaux, France, 2012.

[34] E. Babulak and M. Wang, “Discrete Event Simulation,” *International Journal of Online Engineering,* vol. 4, no. 2, pp. 60-63, 2008.

[35] A. Goti and A. Garcia, “Condition Based Maintenance Optimization of Multi-Equipment Manufacturing Systems by Combining Discrete Event Simulation and Multiobjective Evolutionary Algorithms,” in *Discrete Event Simulations*, Sciyo, 2010, p. 330.

[36] M. A. Majid, U. Aickelin and P.-O. Siebers, “Comparing Simulation Output Accuracy of Discrete Event and Agent Based Models: A Quantitave Approach,” in *SCSC '09 Proceedings of the 2009 Summer Computer Simulation Conference*, Vista, CA, 2009.

[37] P. O. Siebers, C. M. Macal, J. Garnett, D. Buxton and M. Pidds, “Discrete-Event Simulation is Dead, Long Live Agent-Based Simulation!,” *Journal of Simulation,* vol. 4, no. 3, pp. 204-210, 2010.

[38] K.-J. Wang and C.-H. Chou, “Modeling of Agent-Based Manufacturing Systems,” *Journal of the Chinese Institute of Industrial Engineers,* vol. 18, no. 1, pp. 1-14, 2001.

[39] J. April, F. Glover, J. P. Kelly and M. Laguna, “Practical Introduction to Simulation Optimization,” in *Winter Simulation Conference*, Boulder,U.S.A, 2003.

[40] W. Abo-Hamad and A. Arisha, “Simulation Optimisation Methods in Supply Chain Applications: A Review,” *Irish Journal of Management,* 2010.

[41] F. Glover, J. P. Kelly and M. Laguna, “New advances for wedding optimization and simulation,” in *Winter Simulation Conference*, Boulder, 1999.

[42] M. C. Fu, F. W. Glover and J. April, “Simulation optimization: a review, new developments, and applications,” in *Winter Simulation Conference*, Orlando, 2005.

[43] C.-H. Chen and L. H. Lee, “Introduction to Stochastic Simulation Optimization,” in *Stochastic Simulation Optimization: An Optimal Computing Budget Allocation (System Engineering and Operations Research)*, Hackensack, World Scientific Publishing Company, 2010.

[44] Y. Jin and J. Branke, “Evolutionary optimization in uncertain environments-a survey,” *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION,* vol. 9, no. 3, pp. 303 - 317, 2005.

[45] J. E. Fieldsend and R. M. Everson, “Multi-objective optimisation in the presence of uncertainty,” *Evolutionary Computation,* vol. 1, pp. 243 - 250, 2005.

[46] M. C. Fu, “Optimization for simulation: Theory vs. Practice,” *INFORMS Journal on Computing,* vol. 14, no. 3, pp. 192-215, 2002.

[47] OptTek, “How the OptQuest Engine works,” OptTek, [Online]. Available: http://www.opttek.com. [Accessed 28 September 2013].

[48] F. Glover and A. Reinholz, “Metaheuristics in Science and Industry: New Developments,” in *Metaheuristics International Conference*, Montreal ,Canada, June 25-29, 2007.

[49] M. Laguna, “OptQuest: Optimization of Complex Systems,” OPTTEK SYSTEMS,INC., 2011.

[50] R. Martí , M. Laguna and F. Glover, “Principles of scatter search,” *European Journal of Operational Research,* vol. 169, no. 2, pp. 359-372, 2006.

[51] Process Engineering group, “Introduction to SSm,” Instituto de Investigaciones Marinas (C.S.I.C.), Vigo(Spain), 2009.

[52] R. Martí and M. Laguna, “Scatter Search: Basic Design and Advanced Strategies,” *Revista Iberoamericana de Inteligencia Artificial,* vol. 7, no. 19, pp. 123-130, 2003.

[53] I. Boussaïd , J. Lepagnot and P. Siarry , “A survey on optimization metaheuristics,” *Information Sciences,* vol. 237, pp. 82-117, 2013.

[54] F. Glover, M. Laguna and R. Martí , “Fundamentals of Scatter Search and Path Relinking,” *Control and Cybernetics,* vol. 29, no. 3, pp. 653-684, 2000.

[55] K. Deb, Multi-objective Optimization Using Evolutionary Algorithms, Chichester: Wiley-Blackwell, 2001.

[56] M. Gen and R. Cheng, Genetic Algorithms and Engineering Optimization, New York: John Wiley & Sons, 2000.

[57] OptTek, “Multi-Objective Optimization,” OptTek, [Online]. Available: http://www.opttek.com. [Accessed 24 September 2013].

[58] E. J. Hughes, “Multiple Single Objective Pareto Sampling,” *Evolutionary Computation,* vol. 4, pp. 2678-2684, 2003.

[59] T. Sreenuch, A. Tsourdos, E. J. Hughes and B. A. White, “Fuzzy Gain-Scheduled Missile Autopilot Design using Evolutionary Algorithms,” *IEEE Transactions on Aerospace and Electronic Systems,* vol. 42, no. 4, pp. 1323-1339, 2006.

[60] AnyLogic, “Monte Carlo Experiment,” AnyLogic, [Online]. Available: http://www.anylogic.com/. [Accessed 25 October 2013].